

rec'd PCT/PTC 12 APR 2005
P 2004/001541

13. 2. 2004

日 本 国 特 許 庁
JAPAN PATENT OFFICE

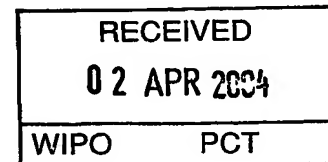
別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 3 年 2 月 1 9 日
Date of Application:

出 願 番 号 特 願 2 0 0 3 - 0 4 0 8 0 4
Application Number:
[ST. 10/C] : [J P 2 0 0 3 - 0 4 0 8 0 4]

出 願 人 松下電器産業株式会社
Applicant(s):

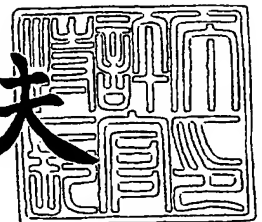


**PRIORITY
DOCUMENT**
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH RULE 17.1(a) OR (b)

2 0 0 4 年 3 月 1 8 日

特許庁長官
Commissioner,
Japan Patent Office

今 井 康 夫



出証番号 出証特 2 0 0 4 - 3 0 2 1 9 3 :

【書類名】 特許願
【整理番号】 2022550053
【あて先】 特許庁長官殿
【国際特許分類】 H04N 7/137

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 安倍 清史

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 角野 眞也

【発明者】

【住所又は居所】 大阪府門真市大字門真 1 0 0 6 番地 松下電器産業株式会社内

【氏名】 近藤 敏志

【特許出願人】

【識別番号】 000005821

【氏名又は名称】 松下電器産業株式会社

【代理人】

【識別番号】 100109210

【弁理士】

【氏名又は名称】 新居 広守

【電話番号】 06-4806-7530

【手数料の表示】

【予納台帳番号】 049515

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 0213583

【ブルーフの要否】 要

【書類名】 明細書

【発明の名称】 動画像の符号化方法および復号化方法

【特許請求の範囲】

【請求項 1】 既に符号化済みのピクチャを参照することによって入力された画像の符号化を行う動画像符号化方法であって、

符号化の途中でピクチャの画像サイズが変更されたかどうかを判定する画像サイズ変更判定ステップを有し、

前記画像サイズ変更判定ステップによって、画像サイズが変更されたと判定されたとき、復号化におけるピクチャメモリ内の全てのメモリ領域を別のデータによる再利用を可能とすることを指示する情報を符号列中に記述する

ことを特徴とする動画像符号化方法。

【請求項 2】 動画像が符号化された信号である符号化信号が記録されたコンピュータ読み取り可能な記録媒体であって、

前記符号化信号には、ピクチャの画像サイズが変更されたときに、復号化におけるピクチャメモリ内の全てのメモリ領域を別のデータによる再利用を可能とすることを指示する情報が含まれている

ことを特徴とする記録媒体。

【請求項 3】 符号化済みピクチャを参照して符号化を行うことによって生成された符号列を復号化する動画像復号化方法であって、

復号化の途中でピクチャの画像サイズが変更されたかどうかを判定する画像サイズ変更判定ステップと、

復号化したピクチャをピクチャメモリに蓄積するときに、既にピクチャメモリに蓄積されているピクチャの中からどのピクチャのメモリ領域を別のデータによる再利用を可能とするかを判定する削除ピクチャ判定ステップとを有し、

前記削除ピクチャ判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、前記画像サイズ変更判定ステップの判定結果に従って、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1枚分もしくは複数枚分のピクチャが蓄積されているメモリ領域を別のデータによる再利用を可能とする

ことを特徴とする動画像復号化方法。

【請求項 4】 前記削除ピクチャ判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1枚分もしくは複数枚分のピクチャが蓄積されているメモリ領域を表示順情報の古いものから順に別のデータによる再利用を可能とする

ことを特徴とする請求項 3 に記載の動画像復号化方法。

【請求項 5】 前記削除ピクチャ判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1枚分もしくは複数枚分のピクチャが蓄積されているメモリ領域を先に復号化されたものから順に別のデータによる再利用を可能とする

ことを特徴とする請求項 3 に記載の動画像復号化方法。

【請求項 6】 既に符号化済みのピクチャを参照することによって入力された画像の符号化を行う動画像符号化方法であって、

復号化におけるピクチャメモリを分割するための情報を符号化するピクチャメモリ分割情報符号化ステップを有する

ことを特徴とする動画像符号化方法。

【請求項 7】 前記ピクチャメモリ分割情報符号化ステップは、復号化におけるピクチャメモリを分割する 1 つ分の領域のサイズとして、符号化対象とするピクチャが持つ可能性のある最大の画像サイズを用いることを示す情報を符号化する

ことを特徴とする請求項 6 に記載の動画像符号化方法。

【請求項 8】 前記ピクチャメモリ分割情報符号化ステップは、復号化におけるピクチャメモリを分割する 1 つ分の領域のサイズとして、符号化対象とするピクチャが持つ可能性のある最小の画像サイズを用いることを示す情報を符号化する

ことを特徴とする請求項 6 に記載の動画像符号化方法。

【請求項 9】 前記ピクチャメモリ分割情報符号化ステップは、復号化にお

けるピクチャメモリを分割する 1 つ分の領域のサイズを直接示す情報を符号化する

ことを特徴とする請求項 6 に記載の動画像符号化方法。

【請求項 10】 前記ピクチャメモリ分割情報符号化ステップは、復号化におけるピクチャメモリを分割する 1 つ分の領域のサイズとして、符号化対象とするピクチャが持つ可能性のある全ての画像サイズに対して横方向 1 列分のマクロブロックのデータが分割されずに 1 つの領域に収められるようなサイズを用いることを示す情報を符号化する

ことを特徴とする請求項 6 に記載の動画像符号化方法。

【請求項 11】 前記ピクチャメモリ分割情報符号化ステップは、前記復号化におけるピクチャメモリを分割するための情報に従ってピクチャメモリを分割することにより、対象とする符号列中の全てのピクチャを問題なく復号化することが可能であることを示す情報を符号化する

ことを特徴とする請求項 6 に記載の動画像符号化方法。

【請求項 12】 前記動画像符号化方法は、

符号化を行うときに参照するピクチャの枚数として、前記ピクチャメモリ分割情報符号化ステップによって分割されて生成される領域の数以下の値を用いる

ことを特徴とする請求項 6 に記載の動画像符号化方法。

【請求項 13】 動画像が符号化された信号である符号化信号が記録されたコンピュータ読み取り可能な記録媒体であって、

前記符号化信号には、復号化におけるピクチャメモリを分割するための情報が含まれている

ことを特徴とする記録媒体。

【請求項 14】 前記符号化信号には、前記復号化におけるピクチャメモリを分割するための情報に従ってピクチャメモリを分割することにより、対象とする符号列中の全てのピクチャを問題なく復号化することが可能であることを示す情報が含まれている

ことを特徴とする請求項 13 に記載の記録媒体。

【請求項 15】 符号化済みピクチャを参照して符号化を行うことによって

生成された符号列を復号化する動画像復号化方法であって、

復号化の途中でピクチャの画像サイズが変更されたかどうかを判定する画像サイズ変更判定ステップと、

符号列中に含まれるピクチャメモリを分割するための情報を示す信号に応じてピクチャメモリを分割するピクチャメモリ分割ステップと、

復号化したピクチャをピクチャメモリに蓄積するときに、既にピクチャデータが格納されているピクチャメモリを分割した領域の中からどの領域を別のデータによる再利用を可能とするかを判定する削除ピクチャ領域判定ステップとを有し

、

前記削除ピクチャ領域判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、前記画像サイズ変更判定ステップの判定結果に従って、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1つもしくは複数の領域に対して別のデータによる再利用を可能とする

ことを特徴とする動画像復号化方法。

【請求項 16】 前記削除ピクチャ領域判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1つもしくは複数の領域を表示順情報の古いピクチャのデータが格納されているものから順に別のデータによる再利用を可能とする

ことを特徴とする請求項 15 に記載の動画像復号化方法。

【請求項 17】 前記削除ピクチャ領域判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1つもしくは複数の領域を先に復号化されたピクチャのデータが格納されているものから順に別のデータによる再利用を可能とする

ことを特徴とする請求項 15 に記載の動画像復号化方法。

【請求項 18】 動画像を符号化するためのプログラムであって、

請求項 1 および 6 から 12 のいずれか 1 項に記載された画像符号化方法に含まれるステップをコンピュータに実行させる

ことを特徴とするプログラム。

【請求項 19】 符号化された動画像の符号列を復号化するためのプログラムであって、

請求項 3 から 5 および 15 から 17 のいずれか 1 項に記載された画像復号化方法に含まれるステップをコンピュータに実行させる ことを特徴とするプログラム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、動画像の符号化方法および復号化方法に関するものであり、特に既に符号化済みのピクチャを参照して画面間予測を行う符号化方法および復号化方法に関するものである。

【0002】

【従来の技術】

近年、音声、画像、その他の画素値を統合的に扱うマルチメディア時代を迎え、従来からの情報メディア、つまり新聞、雑誌、テレビ、ラジオ、電話等の情報を人に伝達する手段がマルチメディアの対象として取り上げられるようになってきた。一般に、マルチメディアとは、文字だけでなく、図形、音声、特に画像等を同時に関連づけて表すことをいうが、上記従来の情報メディアをマルチメディアの対象とするには、その情報をデジタル形式にして表すことが必須条件となる。

【0003】

ところが、上記各情報メディアの持つ情報量をデジタル情報量として見積もってみると、文字の場合 1 文字当たりの情報量は 1 ～ 2 バイトであるのに対し、音声の場合 1 秒当たり 64 kbits (電話品質)、さらに動画については 1 秒当たり 100 Mbits (現行テレビ受信品質) 以上の情報量が必要となり、上記情報メディアでその膨大な情報をデジタル形式でそのまま扱うことは現実的では無い

。例えば、テレビ電話は、64 kbps～1.5 Mbpsの伝送速度を持つサービス総合デジタル網（ISDN: Integrated Services Digital Network）によってすでに実用化されているが、テレビ・カメラの映像をそのままISDNで送ることは不可能である。

【0004】

そこで、必要となってくるのが情報の圧縮技術であり、例えば、テレビ電話の場合、ITU（国際電気通信連合 電気通信標準化部門）で国際標準化されたH.261やH.263規格の動画圧縮技術が用いられている。また、MPEG-1規格の情報圧縮技術によると、通常の音楽用CD（コンパクト・ディスク）に音声情報とともに画像情報を入れることも可能となる。

【0005】

ここで、MPEG（Moving Picture Experts Group）とは、動画面信号のデジタル圧縮の国際規格であり、MPEG-1は、動画面信号を1.5 Mbpsまで、つまりテレビ信号の情報を約100分の1にまで圧縮する規格である。また、MPEG-1規格を対象とする伝送速度が主として約1.5 Mbpsに制限されていることから、さらなる高画質化の要求をみたすべく規格化されたMPEG-2では、動画像信号が2～15 Mbpsに圧縮される。

【0006】

さらに現状では、MPEG-1、MPEG-2と標準化を進めてきた作業グループ（ISO/IEC JTC1/SC29/WG11）によって、より圧縮率が高いMPEG-4が規格化された。MPEG-4では、当初、低ビットレートで効率の高い符号化が可能になるだけでなく、伝送路誤りが発生しても主観的な画質劣化を小さくできる強力な誤り耐性技術も導入されている。また、現在は、ISO/IECとITUの共同で次世代画面符号化方式として、H.264の標準化活動が進んでいる。

【0007】

一般に動画像の符号化では、時間方向および空間方向の冗長性を削減することによって情報量の圧縮を行う。そこで時間的な冗長性の削減を目的とするピクチャ間予測符号化では、前方または後方のピクチャを参照してブロック単位で動き

の検出および予測画像の作成を行い、得られた予測画像と現在のピクチャとの差分値に対して符号化を行う。

【0008】

ここで、ピクチャとは、1枚の画面を表す用語であり、プログレッシブ画像ではフレームを意味し、インタレース画像ではフレームもしくはフィールドを意味する。ここで、インタレース画像とは、1つのフレームが時刻の異なる2つのフィールドから構成される画像である。インタレース画像の符号化や復号化処理においては、1つのフレームをフレームのまま処理したり、2つのフィールドとして処理したり、フレーム内のブロック毎にフレーム構造またはフィールド構造として処理したりすることができる。

【0009】

なお、以下で示すピクチャはプログレッシブ画像でのフレームの意味で説明するが、インタレース画像でのフレームもしくはフィールドであっても同様に説明することができる。

【0010】

図27はピクチャの参照関係を説明するための図であり、(a)は表示順、(b)は符号化順にピクチャを並べた様子を示している。各ピクチャの数字は表示順情報を示し、(a)のように表示順で並べたときに全ての値が昇順になるように割り当てられている。

【0011】

P4を例にとると、P4はP0、P1、P2、P6を参照して符号化もしくは復号化を行っていることが分かる。前記参照に使用される前記複数のピクチャは対象のピクチャP4よりも先に符号化もしくは復号化されている必要があり、ピクチャメモリに参照可能ピクチャとして格納されているものである。(b)のように符号化順に並べるとP4によって参照されるピクチャは全て前方に位置しているのが分かる。符号化および復号化の過程では、ピクチャP4はブロックごとに前記参照に使用される複数のピクチャの中から最適なものを1枚もしくは2枚選択し、選択されたピクチャを参照ピクチャとして後で述べる画面間予測復号化に使用する。

【0012】

また、図27では、P6とP7との間で画面のサイズが変更されている。画面のサイズが変更されたときの最初のピクチャP7はIDRピクチャと呼ばれ次のような特別な性質を持つ。①面内予測復号化のみ可能である。②復号化終了後にピクチャメモリ内の全てのピクチャをリセットする。③画面サイズを変更することが可能である。したがって、IDRピクチャP7の後に復号化されるピクチャはIDRピクチャより先に復号化されたピクチャを参照することができない。そのためP8はP7のみを参照することが可能である。

【0013】

ピクチャメモリに格納された各ピクチャには、参照に使用することができるかできないかを示す情報が付与されている。参照に使用されないピクチャは本来ならばすぐにピクチャメモリから削除するべきであるが、例えば図27のP3は復号化される順番は2番目であるのに対して表示される順番が4番目となっており、たとえ参照に使用されないことが分かっているにもかかわらず表示される順番が来るまで他のピクチャが復号化されるのを待つためにピクチャメモリに格納しておく必要がある。このようなピクチャを表示待ちピクチャまたは参照不可ピクチャと呼ぶ。それに対して参照に使用することができるピクチャを参照可能ピクチャと呼ぶ。

【0014】

IDRピクチャは、スライスデータのヘッダ領域に図26にあるようにreset_flagを有し、前記フラグによって前述したIDRの特徴の1つであるピクチャメモリのリセットの方法が指定される。なお、スライスとは、1つのピクチャを1つもしくは複数の領域に分割した際の分割された1つ1つの領域のことを示すものである。reset_flagが0の場合は、ピクチャメモリ内の全ての参照可能ピクチャを参照不可ピクチャ（表示待ちピクチャ）に変更する。一方、reset_flagが1の場合は、参照可能もしくは参照不可に関わらず、全てのピクチャのデータをピクチャメモリから削除してしまう。つまり、reset_flagが1の場合は、IDRピクチャよりも先に復号化されたピクチャでまだ表示がされていないピクチャがあったとしても、それらピクチャを表示することなくIDRピクチャ以降のピクチャの表示が行われることになる。な

お、削除するとはそのメモリ領域を別のデータの蓄積に使用することを可能とする処理を表すものであり、物理的にデータを消去する場合も、もしくは物理的には消去せずに上書きすることを許可するだけの場合も同様に扱うことが可能である。

【0015】

図24は従来の画像符号化方法を説明するためのブロック図である。この画像符号化方法は、ピクチャメモリ101、予測残差符号化部102、符号列生成部103、予測残差復号化部104、ピクチャメモリ105、動きベクトル検出部106、動き補償符号化部107、動きベクトル記憶部108、ピクチャメモリ制御部109から構成される。

【0016】

符号化対象となる動画画は表示を行う順にピクチャ単位でピクチャメモリ101に入力され、図27(b)に示したような符号化を行う順にピクチャの並び替えを行う。さらに各々のピクチャはマクロブロックと呼ばれる例えば水平16×垂直16画素のブロックに分割されブロック単位で以降の処理が行われる。

【0017】

ピクチャメモリ101から読み出された入力画像信号は差分演算部112に入力され、動き補償符号化部107の出力である予測画像信号との差分を取ることによって得られる差分画像信号を予測残差符号化部102に出力する。予測残差符号化部102では周波数変換、量子化等の画像符号化処理を行い残差信号を出力する。残差信号は予測残差復号化部104に入力され、逆量子化、逆周波数変換等の画像復号化処理を行い残差復号化信号を出力する。加算演算部113では前記残差復号化信号と予測画像信号との加算を行い再構成画像信号を生成し、得られた再構成画像信号はピクチャメモリ105に格納される。

【0018】

一方、ピクチャメモリ101から読み出されたマクロブロック単位の入力画像信号は動きベクトル検出部106にも入力される。ここでは、ピクチャメモリ105に格納されている1枚もしくは複数枚の符号化済みピクチャを探索対象とし、最も入力画像信号に近い画像領域を検出することによってその位置を指し示す

動きベクトルとその際に選択されたピクチャを指し示す参照ピクチャインデックスを決定する。動きベクトル検出はマクロブロックをさらに分割したブロック単位で行われ、得られた動きベクトルは動きベクトル記憶部108に格納される。動き補償符号化部107では、上記処理によって検出された動きベクトルおよび参照ピクチャインデックスを用いて、ピクチャメモリ105に格納されている符号化済みピクチャから最適な画像領域を取り出し予測画像を生成する。

【0019】

上記の一連の処理によって出力された動きベクトル、参照ピクチャインデックス、残差符号化信号等の符号化情報に対して符号列生成部103において可変長符号化を施すことにより、この符号化処理によって出力される符号列が得られる。

【0020】

以上の処理の流れは画面間予測符号化を行った場合の動作であったが、スイッチ114およびスイッチ115によって画面内予測符号化との切り替えがなされる。画面内符号化を行う場合は、動き補償による予測画像の生成は行わず、同一画面内の符号化済み領域から符号化対象領域の予測画像を生成し差分を取ることによって差分画像信号を生成する。差分画像信号は画面間予測符号化の場合と同様に、予測残差符号化部102において残差符号化信号に変換され、符号列生成部103において可変長符号化を施されることにより出力される符号列が得られる。その際に、符号化対象ピクチャがIDRピクチャであった場合はピクチャメモリ制御部109よりピクチャメモリのリセット方法を示す信号が符号列生成部103に入力され前記信号の符号化が行われる。

【0021】

図25は従来の画像復号化方法を説明するためのブロック図である。この画像復号化方法は、符号列解析部201、予測残差復号化部202、ピクチャメモリ203、動き補償復号部204、動きベクトル記憶部205、ピクチャメモリ制御部206から構成される。

【0022】

まず入力された符号列から符号列解析部201によって、動きベクトル、参照

ピクチャインデックス、残差符号化信号等の各種の情報が抽出される。符号列解析器 201 で抽出された動きベクトルは動きベクトル記憶部 205 に、参照ピクチャインデックスは動き補償復号化部 204 に、残差符号化信号は予測残差復号化部 202 にそれぞれ出力される。

【0023】

予測残差復号化部 202 では入力された残差符号化信号に対して、逆量子化、逆周波数変換等の画像復号化処理を施し残差復号化信号を出力する。加算演算部 211 では前記残差復号化信号と動き補償符号化部 204 から出力される予測画像信号との加算を行い再構成画像信号を生成し、得られた再構成画像信号をピクチャメモリ 203 に格納する。

【0024】

動き補償復号化部 204 では、動きベクトル記憶部 205 から入力される動きベクトルおよび符号列解析部 201 から入力される参照ピクチャインデックスを用いて、ピクチャメモリ 203 に格納されている 1 枚もしくは複数枚の復号化済みピクチャから予測画像に最適な画像領域を取り出す。

【0025】

上記の一連の処理によって生成された復号化済みピクチャはピクチャメモリ 203 から表示されるタイミングに従って表示用画像信号として出力される。

以上の処理の流れは画面間予測復号化を行った場合の動作であったが、スイッチ 212 によって画面内予測復号化との切り替えがなされる。画面内復号化を行う場合は、動き補償による予測画像の生成は行わず、同一画面内の復号化済み領域から復号化対象領域の予測画像を生成し加算を行うことによって再構成画像信号を生成し、得られた再構成画像信号はピクチャメモリ 203 に格納され、表示されるタイミングに従って表示用画像信号として出力される。その際に、復号化対象ピクチャが IDR ピクチャであった場合はピクチャメモリ制御部 106 にピクチャメモリのリセット方法を示す信号が入力され、前記信号に基づいてピクチャメモリのリセットを行う。

【0026】

【発明が解決しようとする課題】

従来の技術で説明した復号化方法では、画像のサイズが変更された場合に、ピクチャメモリ内に変更前の画像サイズを持った表示待ちピクチャと変更後の画像サイズを持った参照可能ピクチャおよび表示待ちピクチャとが混在することになるが、画像サイズ変更後のピクチャの保存方法について全く規定がなかった。

【0027】

そこで本発明は、復号化処理において画像サイズが変更された際のピクチャメモリの管理を規定する方法を提案し、任意の画像サイズおよび画像サイズの種類に対して、ピクチャメモリへのアクセス効率を低下させることなく、効率よくピクチャメモリを使用することを目的とする。

【0028】

【課題を解決するための手段】

そしてこの目的を達成するために、本発明による動画像符号化方法は、既に符号化済みのピクチャを参照することによって入力された画像の符号化を行う動画像符号化方法であって、符号化の途中でピクチャの画像サイズが変更されたかどうかを判定する画像サイズ変更判定ステップを有し、前記画像サイズ変更判定ステップによって、画像サイズが変更されたと判定されたとき、復号化におけるピクチャメモリ内の全てのメモリ領域を別のデータによる再利用を可能とすることを指示する情報を符号列中に記述することを特徴とする。

【0029】

さらに、本発明による動画像復号化方法は、符号化済みピクチャを参照して符号化を行うことによって生成された符号列を復号化する動画像復号化方法であって、復号化の途中でピクチャの画像サイズが変更されたかどうかを判定する画像サイズ変更判定ステップと、復号化したピクチャをピクチャメモリに蓄積するときに、既にピクチャメモリに蓄積されているピクチャの中からどのピクチャのメモリ領域を別のデータによる再利用を可能とするかを判定する削除ピクチャ判定ステップとを有し、前記削除ピクチャ判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、前記画像サイズ変更判定ステップの判定結果に従って、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1枚分もしくは複数枚分のピクチャが蓄積されてい

るメモリ領域を別のデータによる再利用を可能とすることを特徴とする。

【0030】

さらに、本発明による別の動画像符号化方法は、既に符号化済みのピクチャを参照することによって入力された画像の符号化を行う動画像符号化方法であって、復号化におけるピクチャメモリを分割するための情報を符号化するピクチャメモリ分割情報符号化ステップを有することを特徴とする。

【0031】

さらに、本発明による別の動画像復号化方法は、符号化済みピクチャを参照して符号化を行うことによって生成された符号列を復号化する動画像復号化方法であって、復号化の途中でピクチャの画像サイズが変更されたかどうかを判定する画像サイズ変更判定ステップと、符号列中に含まれるピクチャメモリを分割するための情報を示す信号に応じてピクチャメモリを分割するピクチャメモリ分割ステップと、復号化したピクチャをピクチャメモリに蓄積するときに、既にピクチャデータが格納されているピクチャメモリを分割した領域の中からどの領域を別のデータによる再利用を可能とするかを判定する削除ピクチャ領域判定ステップとを有し、前記削除ピクチャ領域判定ステップは、ピクチャメモリに復号化したピクチャを蓄積するために十分な空き領域が無かった場合、前記画像サイズ変更判定ステップの判定結果に従って、復号化したピクチャを蓄積するために必要な領域が確保できるまで、1つもしくは複数の領域に対して別のデータによる再利用を可能とすることを特徴とする。

【0032】

【発明の実施の形態】

（実施の形態1）

本発明の実施の形態1における動画像符号化方法について図1に示したブロック図を用いて説明する。符号化処理の構成は図24を用いて説明した従来の方法とほぼ同様である。従来の方法と異なる点は、画面サイズ変更判定部110が追加された点である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0033】

符号化対象のピクチャがIDRピクチャもしくはそれに代わる画像サイズの変更を可能とするピクチャであった場合、画面サイズ変更判定部110では、符号化対象のピクチャの画像サイズとそれ以前に符号化されたピクチャの画像サイズとを比較し、画像サイズの変更がなされたかどうかを判定する。

【0034】

ピクチャメモリ制御部109では、画面サイズ変更判定部110から入力された判定結果が画像サイズが変更されなかったことを示すものであった場合は、通常の符号化と同様の方法を用いて符号化を行う。一方、前記判定結果が画像サイズが変更されたことを示すものであった場合は、復号化におけるピクチャメモリのリセット方法としてピクチャメモリ内の全てのピクチャを参照可能もしくは参照不可に関わらず削除することを示す信号を符号列生成部103に出力する。符号列生成部103では、図26における符号列のreset_flagを1、つまり、ピクチャメモリ内の全てのピクチャを参照可能もしくは参照不可に関わらず削除することを示すようにフラグの値を設定して符号化する。

【0035】

一方、上で述べたような符号化方法によって符号化された符号列の復号化は、図25の従来の方法の説明で用いたブロック図と全く同様の構成を用いて行うことが可能である。従来技術と全く同様の処理についてはここでは説明を省略する。

【0036】

符号列中のスライスヘッダには図26にあるようにreset_flagが符号化されている。符号列解析部201では前記reset_flagの値を解析し、解析結果の情報をピクチャメモリ制御部206に出力する。上で述べたような符号化方法によって生成された符号列では、画像サイズが変更された場合は常にreset_flagが1、つまり、ピクチャメモリ内の全てのピクチャを参照可能もしくは参照不可に関わらず削除することが指示される。これにより、復号化を行う際は、従来方法と全く同様に画像サイズが変更されたかどうかを意識することなく、reset_flagの値に従ってピクチャメモリのリセットを行うことが可能となる。画像サイズが変更されていた場合は、前記reset

__flagの指示によってピクチャメモリの全領域が空き状態になるため、従来どおりに画像サイズ変更後のピクチャを復号化しピクチャメモリに復号化したピクチャを蓄積しても、ピクチャメモリ内に異なる画像サイズのピクチャが混在することは無く、また復号化したピクチャを蓄積する位置も任意の領域を使用することが可能となる。

【0037】

以上の実施の形態で説明したように、符号化を行う際に画面サイズが変更された場合はピクチャメモリ内の全てのピクチャを削除することを示す値を持ったフラグを符号化することにより、復号化側では従来の方法と全く同様の処理を行うことにより矛盾無くピクチャメモリの管理を行うことが可能である。しかし、復号化側のピクチャメモリ内に表示されていないピクチャが残っていた場合、それらのピクチャは表示されることなく削除されてしまうため、画像サイズ変更時に表示画像が時間的に不連続になってしまう可能性がある。

【0038】

(実施の形態2)

本発明の実施の形態2における動画像符号化方法は図24の従来の方法の説明で用いたブロック図と全く同様の構成を用いて行うことが可能である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0039】

符号化対象のピクチャがIDRピクチャもしくはそれに代わる画像サイズの変更を可能とするピクチャであった場合、従来の技術と同様にピクチャメモリ制御部109よりピクチャメモリのリセット方法を示す信号が符号列生成部103に入力される。本実施の形態では、この際に指定するピクチャメモリのリセット方法は任意であり、例えば、復号化側のピクチャメモリの動作を想定した場合に、ピクチャメモリ内に表示されていないピクチャがあるかないかによって、reset__flagを0、つまり全てのピクチャを参照不可にするだけでピクチャデータそのものは削除しない、もしくは、reset__flagを1、つまり全てのピクチャのデータそのものを削除してピクチャメモリの全領域を空き状態にする、かのどちらかを任意に選択することが可能である。符号列生成部103では入

力された前記信号を符号化し、図26のようにreset_flagをスライスヘッダに持った形式の符号列を生成する。

【0040】

次に、上で述べたような符号化方法によって符号化された符号列の復号化方法を、図2に示した復号化処理のブロック図、および図3に示した復号化処理のフローチャート、および図4から図6に示したピクチャメモリの模式図を用いて、まず復号化処理の概要を説明し、その後フローチャートの経路ごとにピクチャメモリ管理に基づいた復号化処理の詳細を説明する。

【0041】

図2に示した復号化処理の構成は図25を用いて説明した従来の方法とほぼ同様である。従来の方法と異なる点は、画面サイズ変更判定部207および削除ピクチャ判定部208が追加された点である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0042】

まず、画像サイズ変更判定部207では、復号化対象のピクチャの画像サイズとそれ以前に復号化されたピクチャの画像サイズとを比較し、画像サイズの変更がなされたかどうかを判定する。

【0043】

符号列中のスライスヘッダには図26にあるようにreset_flagが符号化されている。符号列解析部201では前記reset_flagの値を解析し、解析結果の情報をピクチャメモリ制御部206に出力する。本実施の形態では画面サイズが変更されていた場合でもreset_flagの値は0もしくは1の両方とも選択されている可能性がある。したがって、reset_flagの値に応じて、図3のフローチャートのようにピクチャメモリの管理方法が分かれることになる。特にreset_flagが0の場合は、ピクチャメモリ内に画像サイズ変更前のピクチャが残っている可能性があるため、画像サイズ変更後のピクチャを蓄積するためにどのピクチャを削除しなくてはならないかを判定する必要がある。その判定が削除ピクチャ領域判定部208によってなされ、前記判定結果はピクチャメモリ制御部206に出力されピクチャメモリ管理が行われ

る。

【0044】

次に復号化処理におけるピクチャメモリ制御部206、画像サイズ変更判定部207、削除ピクチャ判定部208の動作の詳細について、図3に示したフローチャートを用いて、それらの経路ごとに説明を行う。なお、図3のフローチャートでは対象ピクチャの復号化を行った後に画像サイズの変更の判定およびreset_flagの値の判定を行っているが、この順番は任意であり、図3に示した順番に限ったものではない。

【0045】

(経路1)

経路1は画像サイズが変更されかつreset_flagが1のときに実行される復号化処理のフローである。reset_flagが1とは、ピクチャメモリ内の全てのピクチャデータを削除し、全ての領域を空き状態にすることを示す値である。つまりこの信号がピクチャメモリ制御部206に入力されると、ピクチャメモリ制御部206はピクチャメモリ内の全てのピクチャを削除し、復号化対象としている画像サイズ変更後のピクチャをピクチャメモリ内の任意の領域に保存することが可能である。

【0046】

(経路2)

経路2は画像サイズが変更されかつreset_flagが0のときに実行される復号化処理のフローである。reset_flagが0とは、ピクチャメモリ内の全てのピクチャデータを参照不可にするがピクチャデータそのものは削除せずに蓄積したままにすることを示す値である。削除ピクチャ判定部208はピクチャメモリ内の空き領域を調べ、空き領域が足りない場合は画像サイズ変更後のサイズと変更前のサイズとを比較し、削除するピクチャの枚数および削除するピクチャを判定する。ピクチャメモリ内に蓄積されているピクチャはピクチャデータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の古いピクチャから順に必要な枚数分だけ削除することが削除ピクチャ判定部208からピクチャメモリ制御部206に伝達される。その結果生成され

た空き領域に復号化したピクチャを蓄積する。

【0047】

図4は経路2による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図4(a-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域があった場合の例である。この例では、図27のピクチャ列におけるP2、P6、P4、P5がピクチャメモリに蓄積され、P7を復号化している状態を示している。この場合は、図4(a-2)のように復号化したピクチャCをそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャの削除は行わない。

【0048】

図4(b-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域がなかった場合の例である。この例では、図27のピクチャ列におけるP3、P1、P2、P6、P4、P5がピクチャメモリに蓄積され、P7を復号化している状態を示している。この場合はピクチャメモリに蓄積されている画像サイズ変更前のピクチャを削除して復号化したピクチャを蓄積するための領域を確保する必要がある。この例では、画像サイズ変更後のピクチャCは画像サイズ変更前のピクチャよりもサイズが大きく、格納するために2枚分のピクチャを削除する必要があることが分かる。次に、削除するピクチャは、各ピクチャが持っている表示順情報が最も古いものから順に選択される。この例では、P1、P2の順に表示順情報が古いことが分かる。削除ピクチャ判定部208では、上記のような処理によって1枚もしくは複数枚の削除するピクチャを判定し、判定結果をピクチャメモリ制御部206に出力しピクチャの削除を行う。図4(b-1)の例では前記処理によってP1、P2が削除され、図4(b-2)のように復号化したピクチャCを生成された空き領域に蓄積している。

【0049】

(経路3)

経路3は画像サイズが変更されずさらにピクチャメモリ内に復号化対象ピクチャと異なる画像サイズを持ったピクチャが存在しなかったときに実行される復号

化処理のフローである。つまり、画像サイズの変更がなされたピクチャが時間的に近くに無い場合、もしくは画像サイズの変更が発生しないストリームを復号化する場合に選択される経路である。削除ピクチャ判定部 208 はピクチャメモリ内の空き領域を調べ、空き領域がない場合は 1 枚分のピクチャ領域を削除するという判定を行う。ピクチャメモリ内に蓄積されているピクチャはピクチャデータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の最も古いピクチャを 1 枚だけ削除することが削除ピクチャ判定部 208 からピクチャメモリ制御部 206 に伝達される。その結果生成された空き領域に復号化したピクチャを蓄積する。

【0050】

図 5 は経路 3 による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図 5 (a-1) はピクチャメモリ内に復号化したピクチャ C を蓄積するのに十分な空き領域があった場合の例である。この例では、図 27 のピクチャ列における P2、P6、P4、P5 がピクチャメモリに蓄積され、P7 を復号化している状態を示している。この場合は、図 5 (a-2) のように復号化したピクチャ C をそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャの削除は行わない。

【0051】

図 5 (b-1) はピクチャメモリ内に復号化したピクチャ C を蓄積するのに十分な空き領域がなかった場合の例である。この例では、図 27 のピクチャ列における P3、P1、P2、P6、P4、P5 がピクチャメモリに蓄積され、P7 を復号化している状態を示している。この場合はピクチャメモリに蓄積されているピクチャを 1 枚だけ削除して復号化したピクチャを蓄積するための領域を確保する必要がある。削除するピクチャの領域は、各ピクチャが持っている表示順情報が最も古いものが選択される。この例では、P1 の表示順情報が古いことが分かる。削除ピクチャ判定部 208 では、上記のような処理によって削除するピクチャの領域を判定し、判定結果をピクチャメモリ制御部 206 に出力しピクチャの削除を行う。図 5 (b-1) の例では前記処理によって P1 が削除され、図 5 (

b-2) のように復号化したピクチャC生成された空き領域に蓄積する。

【0052】

(経路4)

経路4は画像サイズが変更されずさらにピクチャメモリ内に復号化対象ピクチャと異なる画像サイズを持ったピクチャが存在したときに実行される復号化処理のフローである。つまり、画像サイズの変更がなされたピクチャが時間的に近くに有る場合に選択される経路である。削除ピクチャ判定部208はピクチャメモリ内の空き領域を調べ、空き領域が足りない場合は画像サイズ変更後のサイズと変更前のサイズとを比較し、削除するピクチャの枚数および削除するピクチャを判定する。ピクチャメモリ内に蓄積されているピクチャはピクチャデータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の古いピクチャから順に必要な枚数分だけ削除することが削除ピクチャ領域判定部208からピクチャメモリ制御部206に伝達される。その結果生成された空き領域に復号化したピクチャを蓄積する。

【0053】

図6は経路4による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図6(a-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域があった場合の例である。この例では、図27のピクチャ列におけるP4、P5、P7がピクチャメモリに蓄積され、P8を復号化している状態を示している。この場合は、図6(a-2)のように復号化したピクチャCをそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャの削除は行わない。

【0054】

図6(b-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域がなかった場合の例である。この例では、図27のピクチャ列におけるP2、P6、P4、P5、P7がピクチャメモリに蓄積され、P8を復号化している状態を示している。この場合はピクチャメモリに蓄積されている画像サイズ変更前のピクチャを削除して復号化したピクチャを蓄積するための領域を確

保する必要がある。この例では、画像サイズ変更後のピクチャCは画像サイズ変更前のピクチャよりもサイズが大きく、格納するために既に空き領域になっている領域を使用してもさらに2枚分のピクチャを削除する必要があることが分かる。次に、削除するピクチャは、各ピクチャが持っている表示順情報が最も古いものから順に選択される。この例では、P2、P4の順に表示順情報が古いことが分かる。削除ピクチャ判定部208では、上記のような処理によって1枚もしくは複数枚の削除するピクチャの領域を判定し、判定結果をピクチャメモリ制御部206に出力しピクチャの削除を行う。図6(b-1)の例では前記処理によってP2、P4が削除され、図6(b-2)のように復号化したピクチャC生成された空き領域に蓄積する。このとき、P2、P4は物理的に離れた領域に位置しているため、結果的に復号化したピクチャCは物理的に2ヶ所に分かれて蓄積されてしまうことになる。

【0055】

なお、上記実施の形態では、復号化におけるピクチャメモリから1枚もしくは複数枚のピクチャを削除するときに、各ピクチャの持つ表示順情報が古いものから順に行っていたが、表示順情報の代わりに先に復号化されたピクチャから順に削除を行うとした場合も全く同様に扱うことが可能である。

【0056】

以上の実施の形態で説明したように、符号化を行う際に画面サイズが変更された場合にも、表示されていない可能性のあるピクチャをできる限りピクチャメモリに残したまま矛盾無くピクチャメモリの管理を行うことが可能となる。また、ピクチャメモリ内に使用されない無駄な領域がほとんど発生しないため、メモリの有効活用が可能となる。また、符号化側は従来の方法と全く同様の処理によって本実施の形態の処理を実現することが可能であり、符号化装置の作成が容易である。しかし、図6(b-2)の例で発生したように、復号化したピクチャが物理的に分割された領域に蓄積されてしまう状況が発生し、参照ピクチャとして参照に使用する際のメモリのアクセスが複雑になり速度の低下の原因となる可能性がある。特にストリーム中に存在するピクチャの画像サイズの組み合わせによっては、非常に複雑に分割されてしまう可能性がある。

【0057】

(実施の形態3)

本発明の実施の形態3における動画像符号化方法について図7に示したブロック図を用いて説明する。符号化処理の構成は図24を用いて説明した従来の方法とほぼ同様である。従来の方法と異なる点は、ピクチャメモリ分割情報符号化部111が追加された点である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0058】

実施の形態2では復号化におけるピクチャメモリの管理をピクチャ単位で行っていたが、本実施の形態では復号化におけるピクチャメモリを1つもしくは複数の領域に分割し、分割された領域を1つの単位としてピクチャメモリの管理を行うことを特徴とする。このときの分割された領域のことをピクチャ領域と呼ぶ。

【0059】

符号化対象のストリームの符号化に先立って、ピクチャメモリ分割情報符号化部111では復号化側のピクチャメモリを分割する方法を指定するための情報を符号化する。本実施の形態では、前記復号化側のピクチャメモリを分割する方法を指定する情報として、ストリーム中に含まれる可能性のある最大の画像サイズの情報に符号化し、図14(a)における符号列のピクチャ共通情報領域のmax_sizeとして記述する。ここでの最大の画像サイズは、例えば、符号化装置が符号化に使用することのできる最大の画像サイズを用いることや、ユーザが符号化を実行する際に選択して決定することなどが想定される。

【0060】

符号化対象のピクチャがIDRピクチャもしくはそれに代わる画像サイズの変更を可能とするピクチャであった場合、従来の技術と同様にピクチャメモリ制御部109よりピクチャメモリのリセット方法を示す信号が符号列生成部103に入力される。本実施の形態では、この際に指定するピクチャメモリのリセット方法は任意であり、例えば、復号化側のピクチャメモリの動作を想定した場合に、ピクチャメモリ内に表示されていないピクチャがあるかないかによって、reset_flagを0、つまり全てのピクチャを参照不可にするだけでピクチャデー

タそのものは削除しない、もしくは、`reset_flag`を1、つまり全てのピクチャのデータそのものを削除してピクチャメモリの全領域を空き状態にする、かのどちらかを任意に選択することが可能である。符号列生成部103では入力された前記信号を符号化し、図14(a)のように`reset_flag`をスライスヘッダに持った形式の符号列を生成する。

【0061】

なお、`max_size`としてストリーム中に含まれる可能性のある最大の画像サイズの情報を記述する代わりに、図14(b)のように復号化におけるピクチャメモリ内に確保することのできる領域の数を`min_area_num`として符号列のピクチャ共通情報領域に記述することも可能である。その場合、`min_area_num`の値は下記の式1によって決定される。なお、式中の割りの演算結果は切り捨て処理によって整数値として算出されるものとする。

【0062】

$$(\text{min_area_num}) = (\text{復号化側のピクチャメモリのサイズ}) / (\text{max_size}) \quad (\text{式1})$$

なお、上記の画像サイズおよびピクチャメモリのサイズとは、全体の画素数、もしくは全体のマクロブロックの個数、もしくは全体のビット数のいずれであっても同様に扱うことが可能である。

【0063】

なお、画面間予測符号化を行う際に参照することのできるピクチャの枚数を、符号化対象とするピクチャの画面サイズとは無関係に常に上記の式1によって算出される`min_area_num`以下として符号化を行うことにより、復号化側において上記分割方法によってピクチャメモリを分割して使用する場合でも、参照ピクチャがピクチャメモリ内に存在しないという状況が発生することを防ぐことが可能となる。

【0064】

なお、図15に示したように`max_size`および`min_area_num`に加えて`cnf_flag`を符号列のピクチャ共通情報領域に記述することも可能である。`cnf_flag`は`max_size`および`min_area_num`

umに従って復号化側のピクチャメモリを分割して管理を行った場合でも、ストリーム中に含まれる全てのピクチャを問題なく復号化することができるかどうかを示すフラグである。例えば、値が1の場合は問題なく復号化することができることを示し、値が0の場合は本来参照されるピクチャが復号化側のピクチャメモリの領域の数が足りないことにより既にデータを失ってしまったため参照できない等の理由により完全な復号化を行うことができない可能性があることを示す。

【0065】

なお、図14および図15ではmax__sizeおよびmin__area__numおよびcnf__flagをピクチャ共通情報領域に記述しているが、復号化を行う際に必ずしも必要ではないが復号化処理のための補助となるようなデータを集めたデータ領域に記述することも可能である。また、シーケンス全体から参照されるシーケンス共通情報領域に記述することも可能である。また、スライスヘッダに記述することも可能である。

【0066】

次に、上で述べたような符号化方法によって符号化された符号列の復号化方法を、図8に示した復号化処理のブロック図、および図9に示した復号化処理のフローチャート、および図10と図11に示したピクチャメモリの模式図を用いて、まず復号化処理の概要を説明し、その後フローチャートの経路ごとにピクチャメモリ管理に基づいた復号化処理の詳細を説明する。

【0067】

図8に示した復号化処理の構成は図25を用いて説明した従来の方法とほぼ同様である。従来の方法と異なる点は、画面サイズ変更判定部207、削除ピクチャ領域判定部209、ピクチャメモリ分割方法指定部210が追加された点である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0068】

本実施の形態は実施の形態2とは異なり、ピクチャメモリの管理をピクチャ単位で行うのではなくピクチャ領域と呼ばれるピクチャメモリを1つもしくは複数に分割した領域を1つの単位として行うことを特徴とする。そこで、符号列の復号化に先立って、ピクチャメモリ分割方法指定部210ではピクチャメモリの分

割方法を符号列中の信号から解析してピクチャメモリ制御部206に指示する。本実施の形態では、前記ピクチャメモリを分割する方法を指定する情報として、ストリーム中に含まれる可能性のある最大の画像サイズの情報が図14(a)の符号列中に含まれるmax__sizeとしてピクチャ共通情報領域に記述されている。ピクチャメモリ分割方法指定部210は前記信号を解析することによりピクチャメモリの分割方法を決定する。その際に分割される領域の数は下記の式2によって算出される。なお、式中の割り算の演算結果は切り捨て処理によって整数値として算出されるものとする。

【0069】

$$(\text{ピクチャ領域数}) = (\text{復号化装置のピクチャメモリのサイズ}) / (\text{max_size}) \quad (\text{式2})$$

また、図14(a)の符号列の代わりに図14(b)の符号列を用いた場合はピクチャ領域の数は下記の式3によって算出される。

$$(\text{ピクチャ領域数}) = (\text{min_area_num}) \quad (\text{式3})$$

なお、上記の画像サイズおよびピクチャメモリのサイズとは、全体の画素数、もしくは全体のマクロブロックの個数、もしくは全体のビット数のいずれであっても同様に扱うことが可能である。

【0070】

画像サイズ変更判定部207では、復号化対象のピクチャの画像サイズとそれ以前に復号化されたピクチャの画像サイズとを比較し、画像サイズの変更がなされたかどうかを判定する。

【0071】

符号列中のスライスヘッダには図14にあるようにreset__flagが符号化されている。符号列解析部201では前記reset__flagの値を解析し、解析結果の情報をピクチャメモリ制御部206に出力する。本実施の形態では画面サイズが変更されていた場合でもreset__flagの値は0もしくは1の両方とも選択されている可能性がある。したがって、reset__flagの値に応じて、図9のフローチャートのようにピクチャメモリの管理方法が分かれることになる。特にreset__flagが0の場合は、ピクチャメモリ内に

画像サイズ変更前のピクチャが残っている可能性があるため、画像サイズ変更後のピクチャを蓄積するためにどのピクチャ領域を削除しなくてはならないかを判定する必要がある。その判定が削除ピクチャ領域判定部209によってなされ、前記判定結果はピクチャメモリ制御部206に出力されピクチャメモリ管理が行われる。

【0072】

次に復号化処理におけるピクチャメモリ制御部206、画像サイズ変更判定部207、削除ピクチャ領域判定部209の動作の詳細について、図9に示したフローチャートを用いて、それらの経路ごとに説明を行う。なお、図9のフローチャートでは対象ピクチャの復号化を行った後に画像サイズの変更の判定および `reset_flag` の値の判定を行っているが、この順番は任意であり、図9に示した順番に限ったものではない。

【0073】

(経路1)

経路1は画像サイズが変更されかつ `reset_flag` が1のときに実行される復号化処理のフローである。`reset_flag` が1とは、ピクチャメモリ内の全てのピクチャデータを削除し、全ての領域を空き状態にすることを示す値である。つまりこの信号がピクチャメモリ制御部206に入力されると、ピクチャメモリ制御部206はピクチャメモリ内の全ての領域を削除し、復号化を行った画像サイズ変更後のピクチャをピクチャメモリ内の任意の領域に保存することが可能である。

【0074】

(経路2)

経路2は画像サイズが変更されかつ `reset_flag` が0のときに実行される復号化処理のフローである。`reset_flag` が0とは、ピクチャメモリ内の全てのピクチャデータを参照不可にするがピクチャデータそのものは削除せずに蓄積したままにすることを示す値である。削除ピクチャ領域判定部209はピクチャメモリ内の空き領域を調べ、空き領域が足りない場合は削除するピクチャ領域を判定する。ピクチャメモリ内に蓄積されているピクチャはピクチャデ

ータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の最も古いピクチャが蓄積されているピクチャ領域を削除することが削除ピクチャ領域判定部209からピクチャメモリ制御部206に伝達される。その結果生成された空き領域に復号化したピクチャを蓄積する。

【0075】

図10は経路2による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図10(a-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域があった場合の例である。この例では、図27のピクチャ列におけるP4、P5がピクチャメモリに蓄積され、P7を復号化している状態を示している。P4、P5はピクチャ領域と同じ画像サイズを持ったピクチャであり、つまり、シーケンス中に含まれるピクチャの中で最も大きな画像サイズを持ったものであることが分かる。それに対して復号化したピクチャCは、必ずピクチャ領域と同じ画像サイズもしくはそれ以下の画像サイズを持つことになるが、図10(a-1)の例では、P4、P5よりも小さな画像サイズであった場合を示している。この場合は、図10(a-2)のように復号化したピクチャCをそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャデータを持つピクチャ領域の削除は行わない。

【0076】

図10(b-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域がなかった場合の例である。この例では、図27のピクチャ列におけるP6、P4、P5がピクチャメモリに蓄積され、P7を復号化している状態を示している。この場合はピクチャメモリに蓄積されている画像サイズ変更前のピクチャデータを持つピクチャ領域を削除して復号化したピクチャを蓄積するための領域を確保する必要がある。本実施の形態では、復号化を行うピクチャは必ずピクチャ領域と同じ画像サイズもしくはそれ以下の画像サイズを持つものである。従って、ピクチャメモリ内に空き領域が無い場合は、どれか1つのピクチャ領域を削除することによって、復号化したピクチャを蓄積するための領域を確保することができる。削除するピクチャ領域は、蓄積されているピクチャが持つ

ている表示順情報が最も古いものが選択される。この例では、P 4 の表示順情報が最も古いことが分かる。削除ピクチャ領域判定部 209 では、上記のような処理によって削除するピクチャ領域を判定し、判定結果をピクチャメモリ制御部 206 に出力しピクチャ領域の削除を行う。図 10 (b-1) の例では上記処理によって P 4 が削除され、図 10 (b-2) のように復号化したピクチャ C を生成された空き領域に蓄積している。

【0077】

(経路 3)

経路 3 は画像サイズが変更されなかった場合に実行される復号化処理のフローである。本実施の形態は実施の形態 2 とは異なり、ピクチャメモリ内に復号化対象ピクチャと異なる画像サイズを持ったピクチャが存在していてもしていなくても全く同様の処理によってピクチャメモリを管理することが可能である。削除ピクチャ領域判定部 209 はピクチャメモリ内の空き領域を調べ、空き領域が足りない場合は削除するピクチャ領域を判定する。ピクチャメモリ内に蓄積されているピクチャはピクチャデータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の最も古いピクチャが蓄積されているピクチャ領域を削除することが削除ピクチャ領域判定部 209 からピクチャメモリ制御部 206 に伝達される。その結果生成された空き領域に復号化したピクチャを蓄積する。

【0078】

図 11 は経路 3 による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図 11 (a-1) はピクチャメモリ内に復号化したピクチャ C を蓄積するのに十分な空き領域があった場合の例である。この例では、図 27 のピクチャ列における P 5、P 7 がピクチャメモリに蓄積され、P 8 を復号化している状態を示している。P 5 はピクチャ領域と同じ画像サイズを持ったピクチャであり、つまり、シーケンス中に含まれるピクチャの中で最も大きな画像サイズを持ったものであることが分かる。それに対して復号化したピクチャ C は、必ずピクチャ領域と同じ画像サイズもしくはそれ以下の画像サイズを持つことになるが、図 11 (a

－ 1) の例では、P 5 よりも小さな画像サイズであった場合を示している。この場合は、図 1 1 (a-2) のように復号化したピクチャ C をそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャデータを持つピクチャ領域の削除は行わない。

【0079】

図 1 1 (b-1) はピクチャメモリ内に復号化したピクチャ C を蓄積するのに十分な空き領域がなかった場合の例である。この例では、図 2 7 のピクチャ列における P 4、P 5、P 7 がピクチャメモリに蓄積され、P 8 を復号化している状態を示している。この場合はピクチャメモリに蓄積されているピクチャデータを持つピクチャ領域を削除して復号化したピクチャを蓄積するための領域を確保する必要がある。本実施の形態では、復号化を行うピクチャは必ずピクチャメモリを分割した領域であるピクチャ領域と同じ画像サイズもしくはそれ以下の画像サイズを持つものである。従って、ピクチャメモリ内に空き領域が無い場合は、どれか 1 つのピクチャ領域を削除することによって、復号化したピクチャを蓄積するための領域を確保することができる。削除するピクチャ領域は、蓄積されているピクチャが持っている表示順情報が最も古いものが選択される。この例では、P 4 の表示順情報が最も古いことが分かる。削除ピクチャ領域判定部 2 0 9 では、上記のような処理によって削除するピクチャ領域を判定し、判定結果をピクチャメモリ制御部 2 0 6 に出力しピクチャの削除を行う。図 1 1 (b-1) の例では上記処理によって P 4 が削除され、図 1 1 (b-2) のように復号化したピクチャ C を生成された空き領域に蓄積している。

【0080】

なお、上記実施の形態では、復号化におけるピクチャメモリからピクチャ領域を削除するときに、蓄積されているピクチャの持つ表示順情報が古いものから順に行っていたが、表示順情報の代わりに先に復号化されたピクチャが蓄積されているものから順に削除を行うとした場合も全く同様に扱うことが可能である。

【0081】

以上の説明から分かるように、本実施の形態では経路 2 および経路 3 は全く同様の処理によって実現されるものであり、両者を区別する必要がない。つまり、

符号化を行う際に画面サイズが変更された場合でも変更がされなかった場合でも、常に1領域分のみを削除するという単純な処理方法によって、表示されていない可能性のあるピクチャをできる限りピクチャメモリに残したまま矛盾無くピクチャメモリの管理を行うことが可能となる。また、符号化側においても、従来の方法に加えて、ストリーム中に含まれる最大の画像サイズの情報を符号化して符号列に記述するだけで本実施の形態の処理を実現することが可能であり、符号化装置の作成が容易である。しかし、図10および図11の例で発生したように、復号化したピクチャが小さい画像サイズを持つ場合は、それぞれのピクチャ領域中に使用しない無駄な領域が発生するため、ピクチャメモリを有効に活用しきれていないと言いがたい。しかし、どの画像サイズのピクチャも物理的に分割されることなくメモリ上に連続して蓄積することが可能となるため、メモリのアクセスが単純化され速度を低下させることなく復号化を行うことが可能となる。また、通常の画像符号化および復号化においては、画像サイズが変更されても、ピクチャの参照関係（参照に使用するピクチャの枚数等）は変更せずに処理を行うような構成が多く使用される。よって、本実施の形態で説明したような方法によって、効率の良い処理方法による復号化を実現することが可能であると言える。

【0082】

（実施の形態4）

本発明の実施の形態4における動画像符号化方法について図7に示したブロック図を用いて説明する。符号化処理の構成は図24を用いて説明した従来の方法とはほぼ同様である。従来の方法と異なる点は、ピクチャメモリ分割情報符号化部111が追加された点である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0083】

実施の形態2では復号化におけるピクチャメモリの管理をピクチャ単位で行っていたが、本実施の形態では実施の形態3と同様に復号化におけるピクチャメモリを1つもしくは複数の領域に分割し、分割された領域を1つの単位としてピクチャメモリの管理を行うことを特徴とする。このときの分割された領域のことをピクチャ領域と呼ぶ。

【0084】

符号化対象のストリームの符号化に先立って、ピクチャメモリ分割情報符号化部111では復号化側のピクチャメモリを分割する方法を指定するための情報を符号化する。本実施の形態では、前記復号化側のピクチャメモリを分割する方法を指定する情報として、ストリーム中に含まれる可能性のある最小の画像サイズの情報を符号化し、図16(a)における符号列のピクチャ共通情報領域の`min__size`として記述する。ここでの最小の画像サイズは、例えば、符号化装置が符号化に使用することのできる最小の画像サイズを用いることや、ユーザが符号化を実行する際に選択して決定することなどが想定される。

【0085】

符号化対象のピクチャがIDRピクチャもしくはそれに代わる画像サイズの変更を可能とするピクチャであった場合、従来の技術と同様にピクチャメモリ制御部109よりピクチャメモリのリセット方法を示す信号が符号列生成部103に入力される。本実施の形態では、この際に指定するピクチャメモリのリセット方法は任意であり、例えば、復号化側のピクチャメモリの動作を想定した場合に、ピクチャメモリ内に表示されていないピクチャがあるかないかによって、`reset__flag`を0、つまり全てのピクチャを参照不可にするだけでピクチャデータそのものは削除しない、もしくは、`reset__flag`を1、つまり全てのピクチャのデータそのものを削除してピクチャメモリの全領域を空き状態にする、かのどちらかを任意に選択することが可能である。符号列生成部103では入力された前記信号を符号化し、図16(a)のように`reset__flag`をスライスヘッダに持った形式の符号列を生成する。

【0086】

なお、`min__size`としてストリーム中に含まれる可能性のある最小の画像サイズの情報を記述する代わりに、図16(b)のように復号化におけるピクチャメモリを分割した際のそれぞれの分割されたピクチャ領域のサイズを直接`one__mem__size`として記述する方法もある。この`one__mem__size`としては、例えば、ストリーム中に含まれる最小の画像サイズをそのまま用いる方法、ストリーム中に含まれる全ての画像サイズの最大公約数を用いる方法

、復号化側のメモリの動作を想定した際に最も効率良くアクセスすることが可能であるひとまとまりの領域のサイズを用いる方法等が考えられる。

【0087】

なお、`one_mem_size`を決定する際は、1つのマクロブロックを同じ領域に蓄積できるように、もしくは画面の横方向1列分の全てのマクロブロックを同じ領域に蓄積できるように`one_mem_size`の値の切り上げを行うことにより、効率の良いメモリ管理を実現することも可能である。例えば図18のような3種類の画像サイズのピクチャが1つのストリーム内で符号化されているとする。図18(a)は3通りの画像サイズの縦横のマクロブロックの数を示す例である。Aはマクロブロック数が縦6横7、Bはマクロブロック数が縦8横10、Cはマクロブロック数が縦12横12となっている。`one_mem_size`としてマクロブロック48個分のデータを蓄積できる領域のサイズを記述すれば、図18(b)のように、Aの画像サイズのピクチャは1つの領域で、Bの画像サイズのピクチャは2つの領域で、Cの画像サイズのピクチャは3つの領域で1枚分のピクチャのデータを蓄積することが可能となる。このとき横方向1列分は途中で区切られることなく1つの領域に割り当てられているのが分かる。なお、図18の例では全てのピクチャが等分に分割されているが、分割の大きさは不均一でも問題なく扱うことが可能である。

【0088】

なお、`one_mem_size`として復号化側のピクチャメモリを分割した1つ分のピクチャ領域のサイズの情報を記述する代わりに、図16(c)のように分割したピクチャ領域の数を`area_num`として符号列のピクチャ共通情報領域に記述することも可能である。その場合、`area_num`の値は下記の式4によって決定される。なお、式中の割り算の演算結果は切り捨て処理によって整数値として算出されるものとする。

【0089】

$$(\text{area_num}) = (\text{復号化装置のピクチャメモリのサイズ}) / (\text{one_mem_size}) \quad (\text{式4})$$

なお、上記の画像サイズおよびピクチャメモリのサイズとは、全体の画素数、

もしくは全体のマクロブロックの個数、もしくは全体のビット数のいずれであっても同様に扱うことが可能である。

【0090】

なお、図17に示したようにmin__sizeおよびone__mem__sizeおよびarea__numに加えてcnf__flagを符号列のピクチャ共通情報領域に記述することも可能である。cnf__flagはmin__sizeおよびone__mem__sizeおよびarea__numに従って復号化側のピクチャメモリを分割して管理を行った場合でも、ストリーム中に含まれる全てのピクチャを問題なく復号化することができるかどうかを示すフラグである。例えば、値が1の場合は問題なく復号化することができることを示し、値が0の場合は本来参照されるピクチャが復号化側のピクチャメモリの領域の数が足りないことにより既にデータを失ってしまったため参照できない等の理由により完全な復号化を行うことができない可能性があることを示す。

【0091】

なお、図16および図17ではmin__sizeおよびone__mem__sizeおよびarea__numおよびcnf__flagをピクチャ共通情報領域に記述しているが、復号化を行う際に必ずしも必要ではないが復号化処理のための補助となるようなデータを集めたデータ領域に記述することも可能である。また、シーケンス全体から参照されるシーケンス共通情報領域に記述することも可能である。また、スライスヘッダに記述することも可能である。

【0092】

次に、上で述べたような符号化方法によって符号化された符号列の復号化方法を、図8に示した復号化処理のブロック図、および図9に示した復号化処理のフローチャート、および図12と図13に示したピクチャメモリの模式図を用いて、まず復号化処理の概要を説明し、その後フローチャートの経路ごとにピクチャメモリ管理に基づいた復号化処理の詳細を説明する。

【0093】

図8に示した復号化処理の構成は図25を用いて説明した従来の方法とほぼ同様である。従来の方法と異なる点は、画面サイズ変更判定部207、削除ピクチャ

ャ領域判定部 209、ピクチャメモリ分割方法指定部 210 が追加された点である。従来の技術と全く同様の処理についてはここでは説明を省略する。

【0094】

本実施の形態は実施の形態 3 と同様に、ピクチャメモリの管理をピクチャ単位で行うのではなくピクチャ領域と呼ばれるピクチャメモリを 1 つもしくは複数に分割した領域を 1 つの単位として行うことを特徴とする。そこで、符号列の復号化に先立って、ピクチャメモリ分割方法指定部 210 ではピクチャメモリの分割方法を符号列中の信号から解析してピクチャメモリ制御部 206 に指示する。本実施の形態では、前記ピクチャメモリを分割する方法を指定する情報として、ストリーム中に含まれる可能性のある最小の画像サイズの情報図 16 (a) の符号列中に含まれる `min_size` としてピクチャ共通情報領域に記述されている。ピクチャメモリ分割方法指定部 210 は前記信号を解析することによりピクチャメモリの分割方法を決定する。その際に分割される領域の数は下記の式 5 によって算出される。なお、式中の割り算の演算結果は切り捨て処理によって整数値として算出されるものとする。

【0095】

$$(\text{ピクチャ領域数}) = (\text{復号化装置のピクチャメモリのサイズ}) / (\text{min_size}) \quad (\text{式 5})$$

【0096】

また、図 16 (a) の符号列の代わりに図 16 (b) の符号列を用いた場合はピクチャ領域の数は下記の式 6 によって算出される。

$$(\text{ピクチャ領域数}) = (\text{復号化装置のピクチャメモリのサイズ}) / (\text{one_mem_size}) \quad (\text{式 6})$$

【0097】

また、図 16 (a) の符号列の代わりに図 16 (c) の符号列を用いた場合はピクチャ領域の数は下記の式 7 によって算出される。

$$(\text{ピクチャ領域数}) = (\text{area_num}) \quad (\text{式 7})$$

なお、上記の画像サイズおよびピクチャメモリのサイズとは、全体の画素数、もしくは全体のマクロブロックの個数、もしくは全体のビット数のいずれであっ

でも同様に扱うことが可能である。

【0098】

画像サイズ変更判定部207では、復号化対象のピクチャの画像サイズとそれ以前に復号化されたピクチャの画像サイズとを比較し、画像サイズの変更がなされたかどうかを判定する。

【0099】

符号列中のスライスヘッダには図16にあるようにreset_flagが符号化されている。符号列解析部201では前記reset_flagの値を解析し、解析結果の情報をピクチャメモリ制御部206に出力する。本実施の形態では画面サイズが変更されていた場合でもreset_flagの値は0もしくは1の両方とも選択されている可能性がある。したがって、reset_flagの値に応じて、図9のフローチャートのようにピクチャメモリの管理方法が分かれることになる。特にreset_flagが0の場合は、ピクチャメモリ内に画像サイズ変更前のピクチャが残っている可能性があるため、画像サイズ変更後のピクチャを蓄積するためにどのピクチャ領域を削除しなくてはならないかを判定する必要がある。その判定が削除ピクチャ領域判定部209によってなされ、前記判定結果はピクチャメモリ制御部206に出力されピクチャメモリ管理が行われる。

【0100】

次に復号化処理におけるピクチャメモリ制御部206、画像サイズ変更判定部207、削除ピクチャ領域判定部209の動作の詳細について、図9に示したフローチャートを用いて、それらの経路ごとに説明を行う。なお、図9のフローチャートでは対象ピクチャの復号化を行った後に画像サイズの変更の判定およびreset_flagの値の判定を行っているが、この順番は任意であり、図9に示した順番に限ったものではない。

【0101】

(経路1)

経路1は画像サイズが変更されかつreset_flagが1のときに実行される復号化処理のフローである。reset_flagが1とは、ピクチャメモ

り内の全てのピクチャデータを削除し、全ての領域を空き状態にすることを示す値である。つまりこの信号がピクチャメモリ制御部206に入力されると、ピクチャメモリ制御部206はピクチャメモリ内の全ての領域を削除し、復号化を行った画像サイズ変更後のピクチャをピクチャメモリ内の任意の領域に保存することが可能である。

【0102】

(経路2)

経路2は画像サイズが変更されかつreset_flagが0のときに実行される復号化処理のフローである。reset_flagが0とは、ピクチャメモリ内の全てのピクチャデータを参照不可にするがピクチャデータそのものは削除せずに蓄積したままにすることを示す値である。削除ピクチャ領域判定部209はピクチャメモリ内の空き領域を調べ、空き領域が足りない場合は画像サイズ変更後のサイズとピクチャメモリを分割した1つのピクチャ領域のサイズと比較し、復号化したピクチャの分割の方法を決定して、削除するピクチャ領域の数および削除するピクチャ領域を判定する。ピクチャメモリ内に蓄積されているピクチャはピクチャデータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の古いピクチャが蓄積されているピクチャ領域から順に削除することが削除ピクチャ領域判定部209からピクチャメモリ制御部206に伝達される。その結果生成された空き領域に復号化したピクチャを蓄積する。

【0103】

図12は経路2による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図12(a-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域があった場合の例である。この例では、図27のピクチャ列におけるP2、P6、P4、P5がピクチャメモリに蓄積され、P7を復号化している状態を示している。P2、P6、P4、P5はピクチャ領域と同じ画像サイズを持ったピクチャであることが分かる。それに対して復号化したピクチャCは、ピクチャ領域1つ分のサイズよりも大きな画像サイズであり2つに分割して蓄積する例を示している。この場合は、図12(a-2)のように復号化したピクチャ

ャCを2つ分のピクチャ領域を使用してそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャデータを持つピクチャ領域の削除は行わない。

【0104】

図12(b-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域がなかった場合の例である。この例では、図27のピクチャ列におけるP3、P1、P2、P6、P4、P5がピクチャメモリに蓄積され、P7を復号化している状態を示している。この場合はピクチャメモリに蓄積されている画像サイズ変更前のピクチャデータを持つピクチャ領域を削除して復号化したピクチャを蓄積するための領域を確保する必要がある。この例では、画像サイズ変更後のピクチャCはピクチャ領域1つ分のサイズよりも画像サイズが大きく、2つに分割して2つ分のピクチャ領域を使用して蓄積しなくてはならないことが分かる。また、削除するピクチャの領域は、蓄積されているピクチャが持っている表示順情報が古いものから順に選択される。この例では、P1、P2の順に表示順情報が古いことが分かる。削除ピクチャ領域判定部209では、上記のような処理によって削除するピクチャ領域を判定し、判定結果をピクチャメモリ制御部206に出力しピクチャ領域の削除を行う。図12(b-1)の例では前記処理によってP1、P2が削除され、図12(b-2)のように復号化したピクチャCを生成された空き領域に蓄積している。

【0105】

(経路3)

経路3は画像サイズが変更されなかった場合に実行される復号化処理のフローである。本実施の形態は実施の形態2とは異なり、ピクチャメモリ内に復号化対象ピクチャと異なる画像サイズを持ったピクチャが存在していてもしていなくても全く同様の処理によってピクチャメモリを管理することが可能である。削除ピクチャ領域判定部209はピクチャメモリ内の空き領域を調べ、空き領域が足りない場合は画像サイズ変更後のサイズとピクチャメモリを分割した1つのピクチャ領域のサイズと比較し、復号化したピクチャの分割の方法を決定して、削除するピクチャ領域の数および削除するピクチャ領域を判定する。ピクチャメモリ内

に蓄積されているピクチャはピクチャデータに加え表示順の情報を持っている。前記表示順情報を調べることにより、表示順情報の古いピクチャが蓄積されているピクチャ領域から順に削除することが削除ピクチャ領域判定部209からピクチャメモリ制御部206に伝達される。その結果生成された空き領域に復号化したピクチャを蓄積する。

【0106】

図13は経路3による処理に基づいてピクチャメモリの管理を行っている様子の例を示すものである。

図13(a-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域があった場合の例である。この例では、図27のピクチャ列におけるP4、P5、P7がピクチャメモリに蓄積され、P8を復号化している状態を示している。P4、P5はピクチャ領域と同じ画像サイズを持ったピクチャであることが分かる。それに対して復号化したピクチャCは、ピクチャ領域1つ分のサイズよりも多きな画像サイズであり2つに分割して蓄積する例を示している。この場合は、図13(a-2)のように復号化したピクチャCを2つ分のピクチャ領域を使用してそのまま空き領域に蓄積することが可能である。つまり、既にピクチャメモリ内に蓄積されているピクチャデータを持つピクチャ領域の削除は行わない。

【0107】

図13(b-1)はピクチャメモリ内に復号化したピクチャCを蓄積するのに十分な空き領域がなかった場合の例である。この例では、図27のピクチャ列におけるP2、P6、P4、P5、P7がピクチャメモリに蓄積され、P8を復号化している状態を示している。この場合はピクチャメモリに蓄積されているピクチャデータを持つピクチャ領域を削除して復号化したピクチャを蓄積するための領域を確保する必要がある。この例では、画像サイズ変更後のピクチャCはピクチャ領域1つ分のサイズよりも画像サイズが大きく、2つに分割して2つ分のピクチャ領域を使用して蓄積しなくてはならないことが分かる。また、削除するピクチャの領域は、蓄積されているピクチャが持っている表示順情報が古いものから順に選択される。この例では、P2、P4の順に表示順情報が古いことが分か

る。削除ピクチャ領域判定部 209 では、上記のような処理によって削除するピクチャ領域を判定し、判定結果をピクチャメモリ制御部 206 に出力しピクチャ領域の削除を行う。図 13 (b-1) の例では前記処理によって P2、P4 が削除され、図 13 (b-2) のように復号化したピクチャ C を生成された空き領域に蓄積している。

【0108】

なお、上記実施の形態では、復号化におけるピクチャメモリからピクチャ領域を削除するときに、蓄積されているピクチャの持つ表示順情報が古いものから順に行っていたが、表示順情報の代わりに先に復号化されたピクチャが蓄積されているものから順に削除を行うとした場合も全く同様に扱うことが可能である。

【0109】

以上の説明から分かるように、本実施の形態では経路 2 および経路 3 は全く同様の処理によって実現されるものであり、両者を区別する必要がない。つまり、符号化を行う際に画面サイズが変更された場合でも変更がされなかった場合でも、常に復号化したピクチャがいくつ分の領域が必要かを判断し、必要な領域のみを削除するという単純な処理方法によって、表示されていない可能性のあるピクチャをできる限りピクチャメモリに残したまま矛盾無くピクチャメモリの管理を行うことが可能となる。また、符号化側においても、従来の方法に加えて、ストリーム中に含まれる最小の画像サイズの情報、もしくは復号化側を想定してピクチャメモリを分割する 1 つ分のピクチャ領域のサイズを符号化して符号列に記述するだけで本実施の形態の処理を実現することが可能であり、符号化装置の作成が容易である。また、復号化したピクチャを分割するサイズはメモリアクセスの効率を考慮して任意に設定することが可能であり、メモリアクセスに関する速度を低下させることなく復号化を行うことが可能となる。

【0110】

(実施の形態 5)

図 19 は、上記実施の形態の画像符号化方法および画像復号化方法をフレキシブルディスクに格納させたプログラムを用いて、コンピュータシステムにより実施する場合の説明図である。

【0111】

図19 (b) は、フレキシブルディスクの正面からみた外観、断面構造、及びフレキシブルディスクを示し、図19 (a) は、記録媒体本体であるフレキシブルディスクの物理フォーマットの例を示している。フレキシブルディスクFDはケースF内に内蔵され、該ディスクの表面には、同心円状に外周からは内周に向かって複数のトラックTrが形成され、各トラックは角度方向に16のセクタSeに分割されている。従って、上記プログラムを格納したフレキシブルディスクでは、上記フレキシブルディスクFD上に割り当てられた領域に、上記プログラムとしての画像符号化方法および画像復号化方法が記録されている。

【0112】

また、図19 (c) は、フレキシブルディスクFDに上記プログラムの記録再生を行うための構成を示す。上記プログラムをフレキシブルディスクFDに記録する場合は、コンピュータシステムCsから上記プログラムとしての画像符号化方法および画像復号化方法をフレキシブルディスクドライブFDDを介して書き込む。また、フレキシブルディスク内のプログラムにより上記画像符号化方法および画像復号化方法をコンピュータシステム中に構築する場合は、フレキシブルディスクドライブによりプログラムをフレキシブルディスクから読み出し、コンピュータシステムに転送する。

【0113】

なお、上記説明では、記録媒体としてフレキシブルディスクを用いて説明を行ったが、光ディスクを用いても同様に行うことができる。また、記録媒体はこれに限らず、CD-ROM、メモ리카ード、ROMカセット等、プログラムを記録できるものであれば同様に実施することができる。

【0114】

また、図20から図23は、上記実施の形態で示した符号化処理または復号化処理を行う機器、およびこの機器を用いたシステムを説明する図である。

図20は、コンテンツ配信サービスを実現するコンテンツ供給システムex100の全体構成を示すブロック図である。通信サービスの提供エリアを所望の大きさに分割し、各セル内にそれぞれ固定無線局である基地局ex107～ex110が

設置されている。

【0115】

このコンテンツ供給システムex100は、例えば、インターネットex101にインターネットサービスプロバイダex102および電話網ex104、および基地局ex107～ex110を介して、コンピュータex111、PDA (personal digital assistant) ex112、カメラex113、携帯電話ex114、カメラ付きの携帯電話ex115などの各機器が接続される。

【0116】

しかし、コンテンツ供給システムex100は図20のような組合せに限定されず、いずれかを組み合わせて接続するようにしてもよい。また、固定無線局である基地局ex107～ex110を介さずに、各機器が電話網ex104に直接接続されてもよい。

【0117】

カメラex113はデジタルビデオカメラ等の動画撮影が可能な機器である。また、携帯電話は、PDC (Personal Digital Communications) 方式、CDMA (Code Division Multiple Access) 方式、W-CDMA (Wideband-Code Division Multiple Access) 方式、若しくはGSM (Global System for Mobile Communications) 方式の携帯電話機、またはPHS (Personal Handyphone System) 等であり、いずれでも構わない。

【0118】

また、ストリーミングサーバex103は、カメラex113から基地局ex109、電話網ex104を通じて接続されており、カメラex113を用いてユーザが送信する符号化処理されたデータに基づいたライブ配信等が可能になる。撮影したデータの符号化処理はカメラex113で行っても、データの送信処理をするサーバ等で行ってもよい。また、カメラex116で撮影した動画データはコンピュータex111を介してストリーミングサーバex103に送信されてもよい。カメラex116はデジタルカメラ等の静止画、動画が撮影可能な機器である。この場合、動画データの符号化はカメラex116で行ってもコンピュータex111で行ってもどちらでもよい。また、符号化処理はコンピュータex111やカメラex116

が有するLSIex117において処理することになる。なお、画像符号化・復号化用のソフトウェアをコンピュータex111等で読み取り可能な記録媒体である何らかの蓄積メディア（CD-ROM、フレキシブルディスク、ハードディスクなど）に組み込んでもよい。さらに、カメラ付きの携帯電話ex115で動画データを送信してもよい。このときの動画データは携帯電話ex115が有するLSIで符号化処理されたデータである。

【0119】

このコンテンツ供給システムex100では、ユーザがカメラex113、カメラex116等で撮影しているコンテンツ（例えば、音楽ライブを撮影した映像等）を上記実施の形態同様に符号化処理してストリーミングサーバex103に送信する一方で、ストリーミングサーバex103は要求のあったクライアントに対して上記コンテンツデータをストリーム配信する。クライアントとしては、上記符号化処理されたデータを復号化することが可能な、コンピュータex111、PDAex112、カメラex113、携帯電話ex114等がある。このようにすることでコンテンツ供給システムex100は、符号化されたデータをクライアントにおいて受信して再生することができ、さらにクライアントにおいてリアルタイムで受信して復号化し、再生することにより、個人放送をも実現可能になるシステムである。

【0120】

このシステムを構成する各機器の符号化、復号化には上記各実施の形態で示した動画像符号化方法あるいは動画像復号化方法を用いるようにすればよい。

その一例として携帯電話について説明する。

【0121】

図21は、上記実施の形態で説明した動画像符号化方法と動画像復号化方法を用いた携帯電話ex115を示す図である。携帯電話ex115は、基地局ex110との間で電波を送受信するためのアンテナex201、CCDカメラ等の映像、静止画を撮ることが可能なカメラ部ex203、カメラ部ex203で撮影した映像、アンテナex201で受信した映像等が復号化されたデータを表示する液晶ディスプレイ等の表示部ex202、操作キーex204群から構成される本体部、音声

出力をするためのスピーカ等の音声出力部ex208、音声入力をするためのマイク等の音声入力部ex205、撮影した動画もしくは静止画のデータ、受信したメールのデータ、動画のデータもしくは静止画のデータ等、符号化されたデータまたは復号化されたデータを保存するための記録メディアex207、携帯電話ex115に記録メディアex207を装着可能とするためのスロット部ex206を有している。記録メディアex207はSDカード等のプラスチックケース内に電氣的に書換えや消去が可能な不揮発性メモリであるEEPROM (Electrically Erasable and Programmable Read Only Memory) の一種であるフラッシュメモリ素子を格納したものである。

【0122】

さらに、携帯電話ex115について図22を用いて説明する。携帯電話ex115は表示部ex202及び操作キーex204を備えた本体部の各部を統括的に制御するようになされた主制御部ex311に対して、電源回路部ex310、操作入力制御部ex304、画像符号化部ex312、カメラインターフェース部ex303、LCD (Liquid Crystal Display) 制御部ex302、画像復号化部ex309、多重分離部ex308、記録再生部ex307、変復調回路部ex306及び音声処理部ex305が同期バスex313を介して互いに接続されている。

【0123】

電源回路部ex310は、ユーザの操作により終話及び電源キーがオン状態にされると、バッテリーパックから各部に対して電力を供給することによりカメラ付デジタル携帯電話ex115を動作可能な状態に起動する。

【0124】

携帯電話ex115は、CPU、ROM及びRAM等である主制御部ex311の制御に基づいて、音声通話モード時に音声入力部ex205で集音した音声信号を音声処理部ex305によってデジタル音声データに変換し、これを変復調回路部ex306でスペクトラム拡散処理し、送受信回路部ex301でデジタルアナログ変換処理及び周波数変換処理を施した後にアンテナex201を介して送信する。また携帯電話機ex115は、音声通話モード時にアンテナex201で受信した受信信号を増幅して周波数変換処理及びアナログデジタル変換処理を施し、

変復調回路部ex306でスペクトラム逆拡散処理し、音声処理部ex305によってアナログ音声信号に変換した後、これを音声出力部ex208を介して出力する。

【0125】

さらに、データ通信モード時に電子メールを送信する場合、本体部の操作キーex204の操作によって入力された電子メールのテキストデータは操作入力制御部ex304を介して主制御部ex311に送出される。主制御部ex311は、テキストデータを変復調回路部ex306でスペクトラム拡散処理し、送受信回路部ex301でデジタルアナログ変換処理及び周波数変換処理を施した後にアンテナex201を介して基地局ex110へ送信する。

【0126】

データ通信モード時に画像データを送信する場合、カメラ部ex203で撮像された画像データをカメラインターフェース部ex303を介して画像符号化部ex312に供給する。また、画像データを送信しない場合には、カメラ部ex203で撮像した画像データをカメラインターフェース部ex303及びLCD制御部ex302を介して表示部ex202に直接表示することも可能である。

【0127】

画像符号化部ex312は、本願発明で説明した画像符号化方法を備えた構成であり、カメラ部ex203から供給された画像データを上記実施の形態で示した画像符号化方法によって圧縮符号化することにより符号化画像データに変換し、これを多重分離部ex308に送出する。また、このとき同時に携帯電話機ex115は、カメラ部ex203で撮像中に音声入力部ex205で集音した音声を音声処理部ex305を介してデジタルの音声データとして多重分離部ex308に送出する。

【0128】

多重分離部ex308は、画像符号化部ex312から供給された符号化画像データと音声処理部ex305から供給された音声データとを所定の方式で多重化し、その結果得られる多重化データを変復調回路部ex306でスペクトラム拡散処理し、送受信回路部ex301でデジタルアナログ変換処理及び周波数変換処理を

施した後にアンテナex201を介して送信する。

【0129】

データ通信モード時にホームページ等にリンクされた動画像ファイルのデータを受信する場合、アンテナex201を介して基地局ex110から受信した受信信号を変復調回路部ex306でスペクトラム逆拡散処理し、その結果得られる多重化データを多重分離部ex308に送出する。

【0130】

また、アンテナex201を介して受信された多重化データを復号化するには、多重分離部ex308は、多重化データを分離することにより画像データの符号化ビットストリームと音声データの符号化ビットストリームとに分け、同期バスex313を介して当該符号化画像データを画像復号化部ex309に供給すると共に当該音声データを音声処理部ex305に供給する。

【0131】

次に、画像復号化部ex309は、本願発明で説明した画像復号化方法を備えた構成であり、画像データの符号化ビットストリームを上記実施の形態で示した符号化方法に対応した復号化方法で復号することにより再生動画像データを生成し、これをLCD制御部ex302を介して表示部ex202に供給し、これにより、例えばホームページにリンクされた動画像ファイルに含まれる動画データが表示される。このとき同時に音声処理部ex305は、音声データをアナログ音声信号に変換した後、これを音声出力部ex208に供給し、これにより、例えばホームページにリンクされた動画像ファイルに含まる音声データが再生される。

【0132】

なお、上記システムの例に限られず、最近では衛星、地上波によるデジタル放送が話題となっており、図23に示すようにデジタル放送用システムにも上記実施の形態の少なくとも画像符号化方法または画像復号化方法のいずれかを組み込むことができる。具体的には、放送局ex409では映像情報の符号化ビットストリームが電波を介して通信または放送衛星ex410に伝送される。これを受けた放送衛星ex410は、放送用の電波を発信し、この電波を衛星放送受信設備をもつ家庭のアンテナex406で受信し、テレビ（受信機）ex401またはセット

トップボックス (STB) ex 4 0 7 などの装置により符号化ビットストリームを復号化してこれを再生する。また、記録媒体であるCDやDVD等の蓄積メディア ex 4 0 2 に記録した符号化ビットストリームを読み取り、復号化する再生装置 ex 4 0 3 にも上記実施の形態で示した画像復号化方法を実装することが可能である。この場合、再生された映像信号はモニタ ex 4 0 4 に表示される。また、ケーブルテレビ用のケーブル ex 4 0 5 または衛星/地上波放送のアンテナ ex 4 0 6 に接続されたセットトップボックス ex 4 0 7 内に画像復号化方法を実装し、これをテレビのモニタ ex 4 0 8 で再生する構成も考えられる。このときセットトップボックスではなく、テレビ内に画像復号化方法を組み込んでも良い。また、アンテナ ex 4 1 1 を有する車 ex 4 1 2 で衛星 ex 4 1 0 からまたは基地局 ex 1 0 7 等から信号を受信し、車 ex 4 1 2 が有するカーナビゲーション ex 4 1 3 等の表示装置に動画を再生することも可能である。

【0133】

更に、画像信号を上記実施の形態で示した画像符号化方法で符号化し、記録媒体に記録することもできる。具体例としては、DVDディスク ex 4 2 1 に画像信号を記録するDVDレコーダや、ハードディスクに記録するディスクレコーダなどのレコーダ ex 4 2 0 がある。更にSDカード ex 4 2 2 に記録することもできる。レコーダ ex 4 2 0 が上記実施の形態で示した画像復号化方法を備えていれば、DVDディスク ex 4 2 1 やSDカード ex 4 2 2 に記録した画像信号を再生し、モニタ ex 4 0 8 で表示することができる。

【0134】

なお、カーナビゲーション ex 4 1 3 の構成は例えば図 2 2 に示す構成のうち、カメラ部 ex 2 0 3 とカメラインターフェース部 ex 3 0 3、画像符号化部 ex 3 1 2 を除いた構成が考えられ、同様なことがコンピュータ ex 1 1 1 やテレビ (受信機) ex 4 0 1 等でも考えられる。

【0135】

また、上記携帯電話 ex 1 1 4 等の端末は、符号化器・復号化器を両方持つ送受信型の端末の他に、符号化器のみの送信端末、復号化器のみの受信端末の3通りの実装形式が考えられる。

【0136】

このように、上記実施の形態で示した動画像符号化方法あるいは動画像復号化方法を上述したいずれの機器・システムに用いることは可能であり、そうすることで、上記実施の形態で説明した効果を得ることができる。

【0137】**【発明の効果】**

以上のように、実施の形態1で説明した方法を用いると、符号化を行う際に画面サイズが変更された場合はピクチャメモリ内のピクチャを全て削除することを示す値を持ったフラグを符号化することにより、復号化側では従来の方法と全く同様の処理を行うことにより、矛盾無くピクチャメモリの管理を行うことが可能である。しかし、復号化側のピクチャメモリ内に表示されていないピクチャが残っていた場合、それらのピクチャは表示されることなく削除されてしまうため、画像サイズ変更時に表示画像が時間的に不連続になってしまう可能性がある。

【0138】

また、実施の形態2で説明した方法を用いると、符号化を行う際に画面サイズが変更された場合にも、表示されていない可能性のあるピクチャをできる限りピクチャメモリに残したまま矛盾無くピクチャメモリの管理を行うことが可能となる。また、ピクチャメモリ内に使用されない無駄な領域がほとんど発生しないため、メモリの有効活用が可能となる。また、符号化側は従来の方法と全く同様の処理によって本実施の形態の処理を実現することが可能であり、符号化装置の作成が容易である。しかし、復号化したピクチャが物理的に分割された領域に蓄積されてしまう状況が発生し、参照ピクチャとして参照に使用する際のメモリのアクセスが複雑になり速度の低下の原因となる可能性がある。特にストリーム中に存在するピクチャの画像サイズの組み合わせによっては、非常に複雑に分割されてしまう可能性がある。

【0139】

また、実施の形態3で説明した方法を用いると、符号化を行う際に画面サイズが変更された場合でも変更がされなかった場合でも、常に1領域分のみを削除するという単純な処理方法によって、表示されていない可能性のあるピクチャをで

きる限りピクチャメモリに残したまま矛盾無くピクチャメモリの管理を行うことが可能となる。また、符号化側においても、従来の方法に加えて、ストリーム中に含まれる最大の画像サイズの情報を符号化して符号列に記述するだけで本実施の形態の処理を実現することが可能であり、符号化装置の作成が容易である。しかし、復号化したピクチャが小さい画像サイズを持つ場合は、それぞれのピクチャ領域中に使用しない無駄な領域が発生するため、ピクチャメモリを有効に活用しきれているとはいいがたい。しかし、どの画像サイズのピクチャも物理的に分割されることなくメモリ上に連続して蓄積することが可能となるため、メモリのアクセスが単純化され速度を低下させることなく復号化を行うことが可能となる。また、通常の画像符号化および復号化においては、画像サイズが変更されても、ピクチャの参照関係（参照に使用するピクチャの枚数等）は変更せずに処理を行うような構成が多く使用される。よって、本実施の形態で説明したような方法によって、効率の良い処理方法による復号化を実現することが可能であると言える。

【0140】

また、実施の形態4で説明した方法を用いると、符号化を行う際に画面サイズが変更された場合でも変更がされなかった場合でも、常に復号化したピクチャがいくつかの領域が必要かを判断し、必要な領域のみを削除するという単純な処理方法によって、表示されていない可能性のあるピクチャをできる限りピクチャメモリに残したまま矛盾無くピクチャメモリの管理を行うことが可能となる。また、符号化側においても、従来の方法に加えて、ストリーム中に含まれる最小の画像サイズの情報、もしくは復号化側を想定してピクチャメモリを分割する1つ分のピクチャ領域のサイズを符号化して符号列に記述するだけで本実施の形態の処理を実現することが可能であり、符号化装置の作成が容易である。また、復号化したピクチャを分割するサイズはメモリアクセスの効率を考慮して任意に設定することが可能であり、メモリアクセスに関する速度を低下させることなく復号化を行うことが可能となる。

【図面の簡単な説明】

【図1】

本発明の実施の形態 1 による符号化の動作を説明するためのブロック図である。

【図 2】

本発明の実施の形態 2 による復号化の動作を説明するためのブロック図である。

【図 3】

本発明の実施の形態 2 における復号化処理のピクチャメモリ管理の流れを説明するためのフローチャートである。

【図 4】

本発明の実施の形態 2 における復号化におけるピクチャメモリ管理処理の流れの経路 2 を説明するためのピクチャメモリの概念図である。

【図 5】

本発明の実施の形態 2 における復号化におけるピクチャメモリ管理処理の流れの経路 3 を説明するためのピクチャメモリの概念図である。

【図 6】

本発明の実施の形態 2 における復号化におけるピクチャメモリ管理処理の流れの経路 4 を説明するためのピクチャメモリの概念図である。

【図 7】

本発明の実施の形態 3 および実施の形態 4 による符号化の動作を説明するためのブロック図である。

【図 8】

本発明の実施の形態 3 および実施の形態 4 による復号化の動作を説明するためのブロック図である。

【図 9】

本発明の実施の形態 3 および実施の形態 4 における復号化処理のピクチャメモリ管理の流れを説明するためのフローチャートである。

【図 10】

本発明の実施の形態 3 における復号化におけるピクチャメモリ管理処理の流れの経路 2 を説明するためのピクチャメモリの概念図である。

【図 1 1】

本発明の実施の形態 3 における復号化におけるピクチャメモリ管理処理の流れの経路 3 を説明するためのピクチャメモリの概念図である。

【図 1 2】

本発明の実施の形態 4 における復号化におけるピクチャメモリ管理処理の流れの経路 2 を説明するためのピクチャメモリの概念図である。

【図 1 3】

本発明の実施の形態 4 における復号化におけるピクチャメモリ管理処理の流れの経路 3 を説明するためのピクチャメモリの概念図である。

【図 1 4】

本発明の実施の形態 3 における符号列の構成を説明するための模式図である。

【図 1 5】

本発明の実施の形態 3 における別の符号列の構成を説明するための模式図である。

【図 1 6】

本発明の実施の形態 4 における符号列の構成を説明するための模式図である。

【図 1 7】

本発明の実施の形態 4 における別の符号列の構成を説明するための模式図である。

【図 1 8】

画像サイズの異なるピクチャを分割する方法を説明するための模式図である。

【図 1 9】

上記各実施の形態の動画像の符号化方法および復号化方法をコンピュータシステムにより実現するためのプログラムを格納するための記録媒体についての説明図である。

【図 2 0】

コンテンツ供給システムの全体構成を示すブロック図である。

【図 2 1】

携帯電話の外観図である。

【図 2 2】

携帯電話の構成を示すブロック図である。

【図 2 3】

デジタル放送用システムの例を示す図である。

【図 2 4】

従来の技術および実施の形態 2 における符号化の動作を説明するためのブロック図である。

【図 2 5】

従来の技術および実施の形態 1 における復号化の動作を説明するためのブロック図である。

【図 2 6】

従来の符号列の構成を説明するための模式図である。

【図 2 7】

ピクチャの参照関係を説明するための概念図である。

【符号の説明】

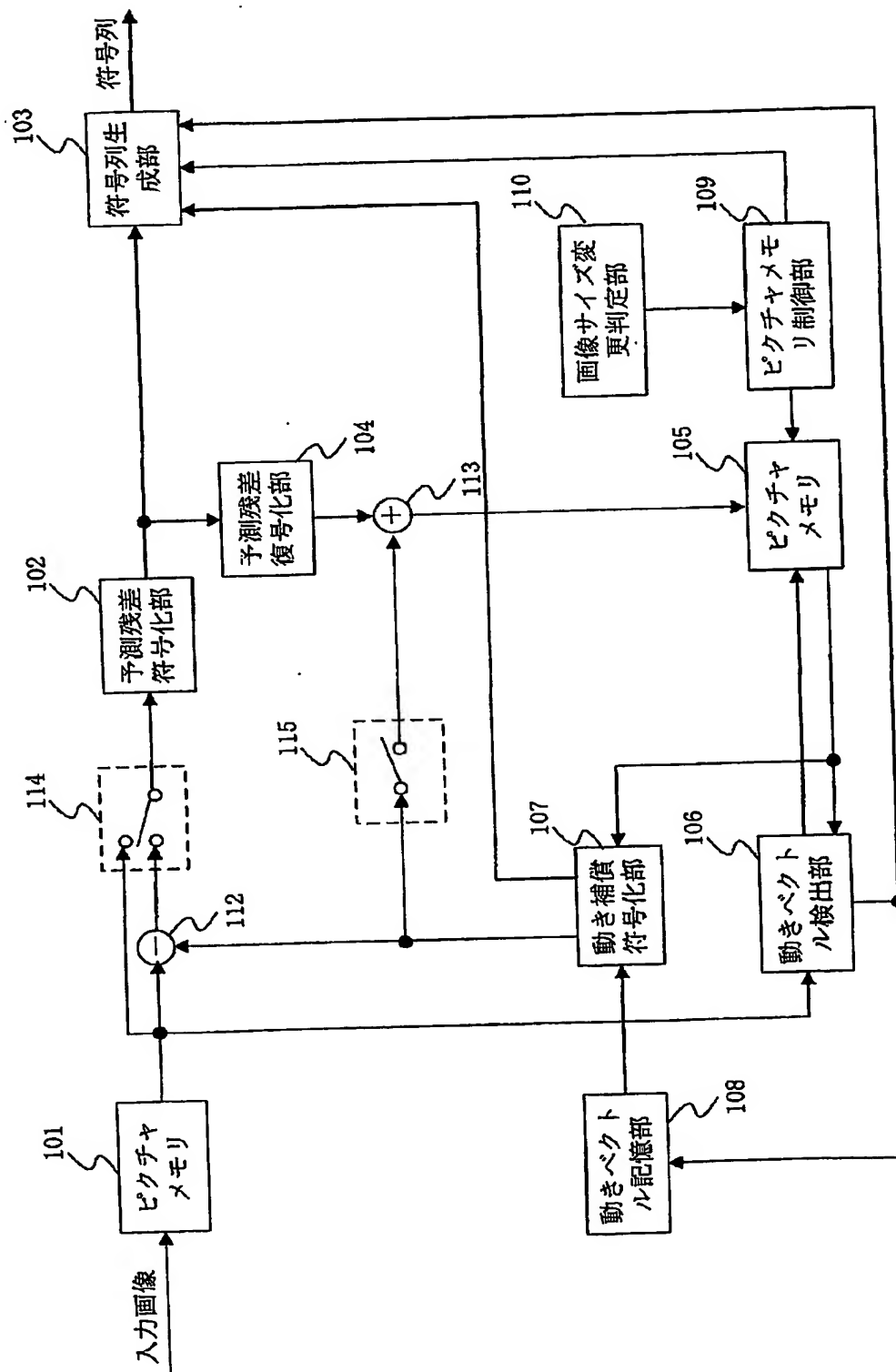
- 101 ピクチャメモリ
- 102 予測残差符号化部
- 103 符号列生成部
- 104 予測残差復号化部
- 105 ピクチャメモリ
- 106 動きベクトル検出部
- 107 動き補償符号化部
- 108 動きベクトル記憶部
- 109 ピクチャメモリ制御部
- 110 画像サイズ変更判定部
- 111 ピクチャメモリ分割情報符号化部
- 201 符号列解析部
- 202 予測残差復号化部
- 203 ピクチャメモリ

- 204 動き補償復号化部
- 205 動きベクトル記憶部
- 206 ピクチャメモリ制御部
- 207 画像サイズ変更判定部
- 208 削除ピクチャ判定部
- 209 削除ピクチャ領域判定部
- 210 ピクチャメモリ分割方法指定部
- Cs コンピュータシステム
- FD フレキシブルディスク
- FDD フレキシブルディスクドライブ

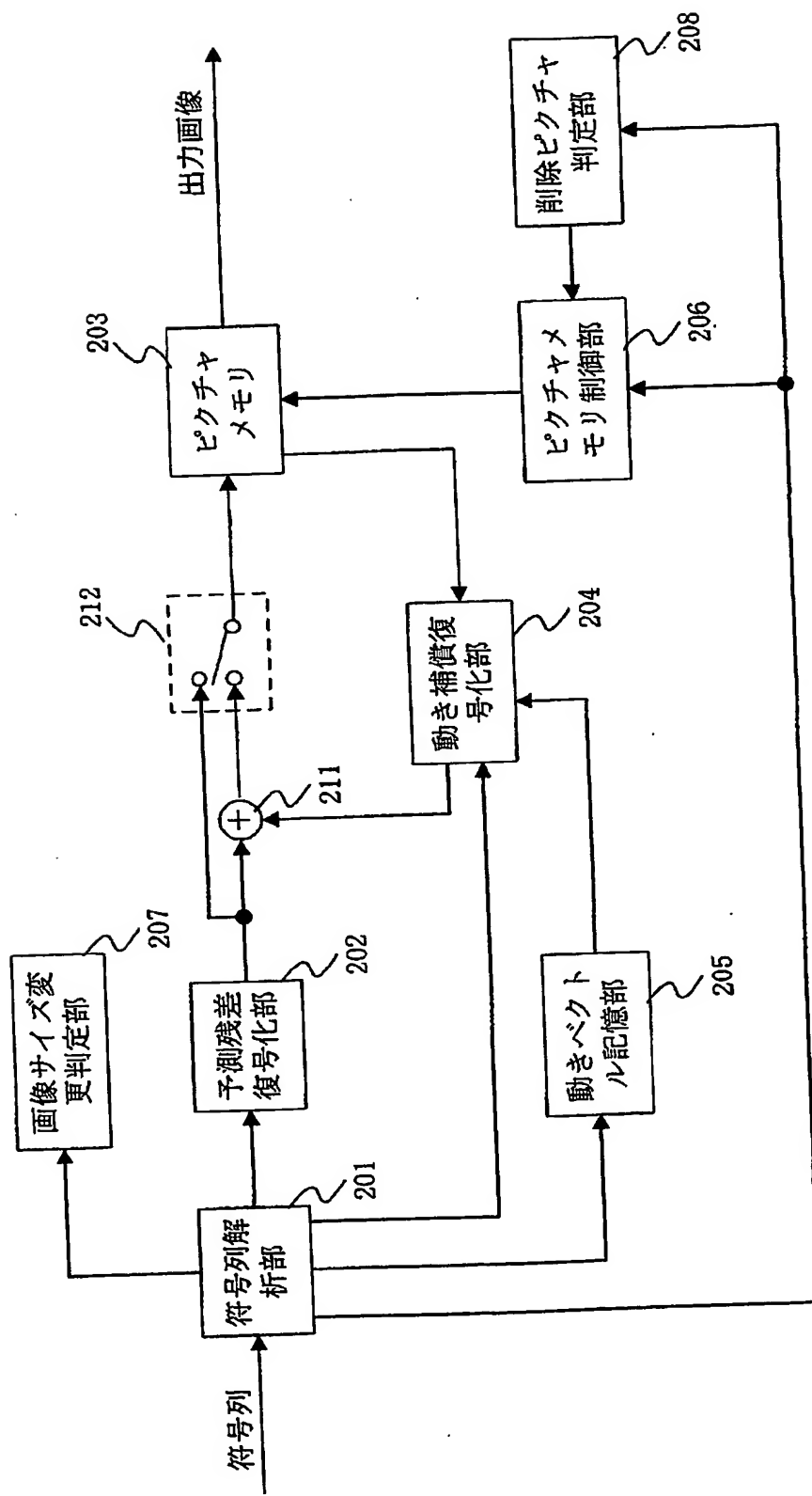
【書類名】

凶面

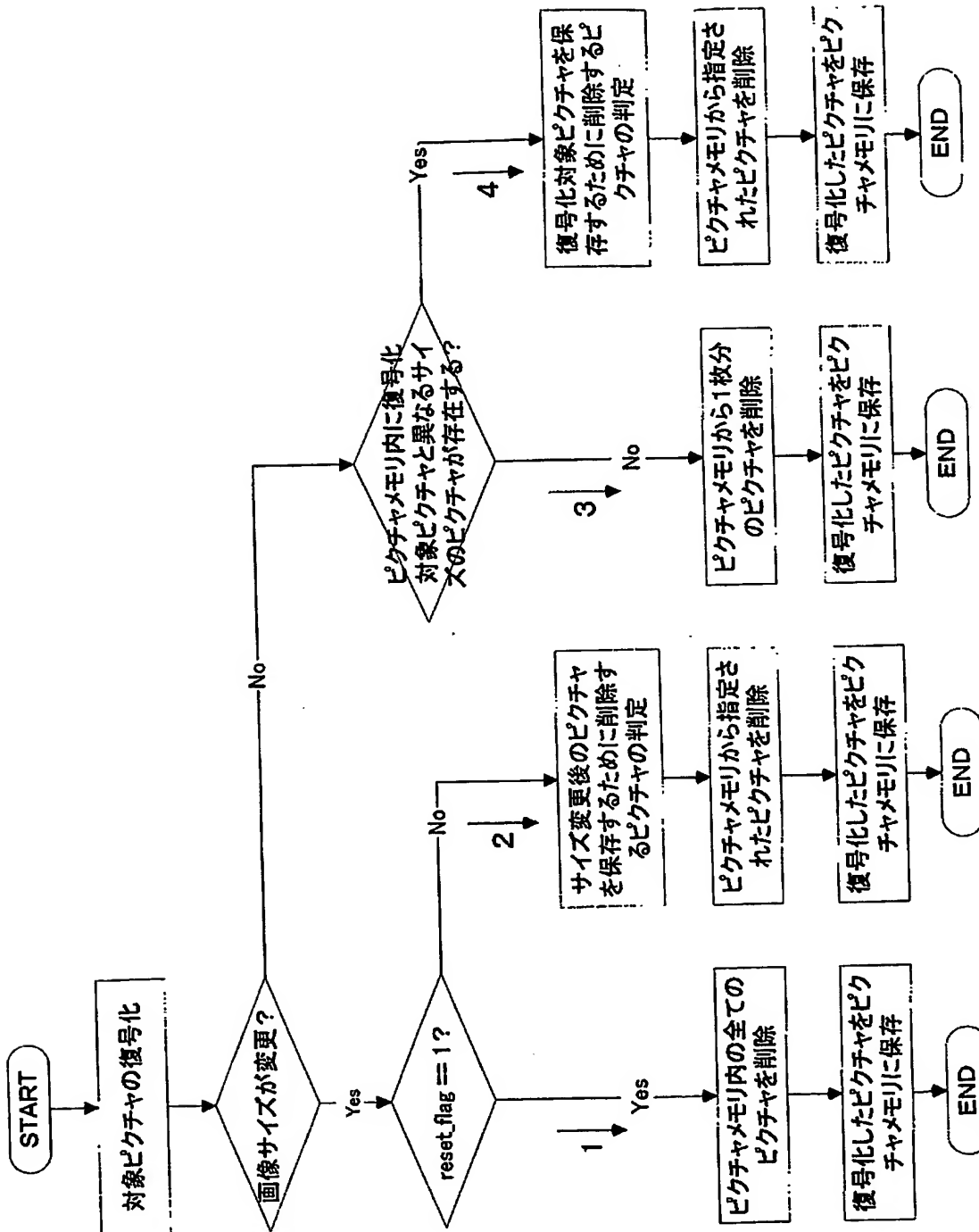
【図 1】



【図2】

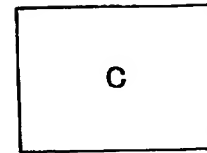
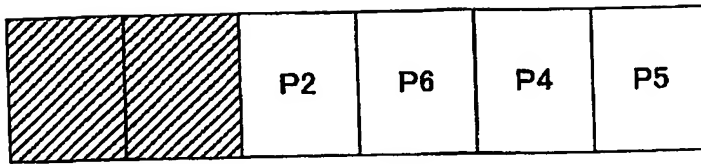


【図 3】

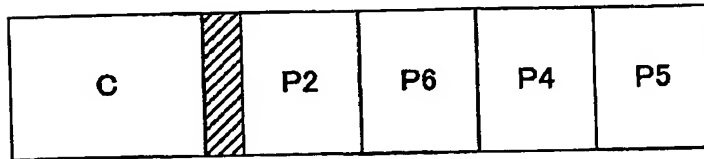


【図 4】

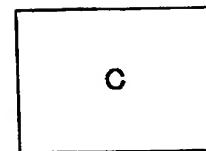
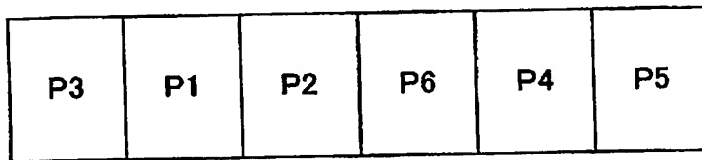
(a-1)



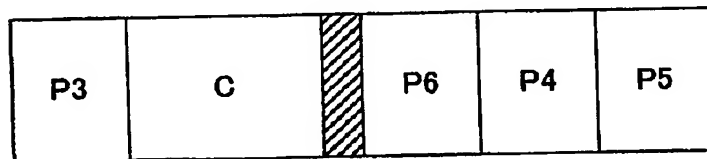
(a-2)



(b-1)

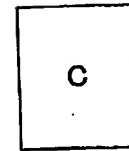
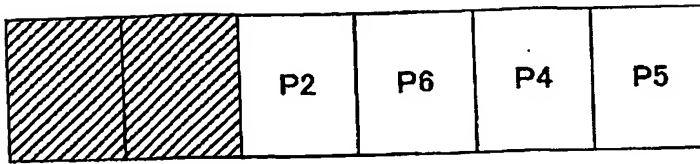


(b-2)

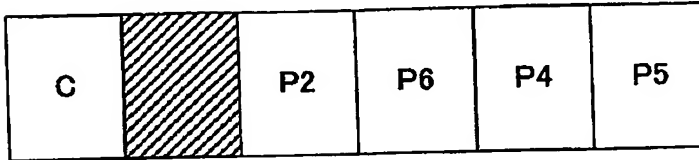


【図5】

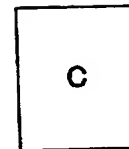
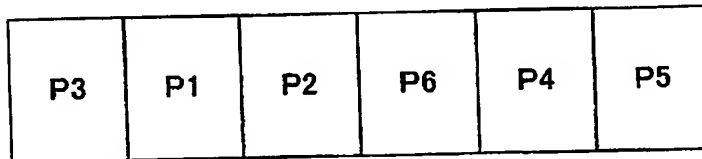
(a-1)



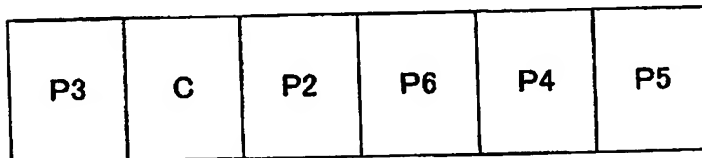
(a-2)



(b-1)

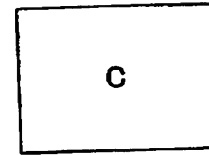
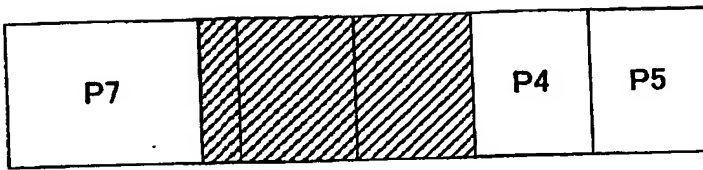


(b-2)

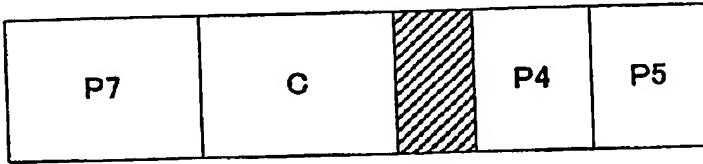


【図 6】

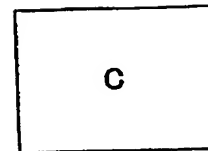
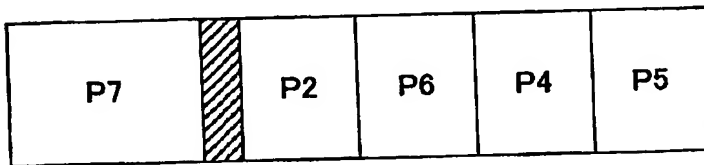
(a-1)



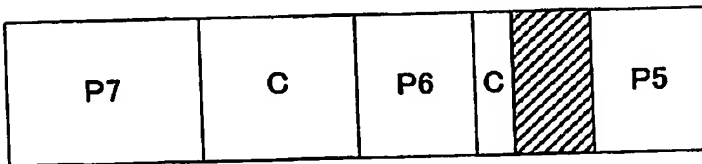
(a-2)



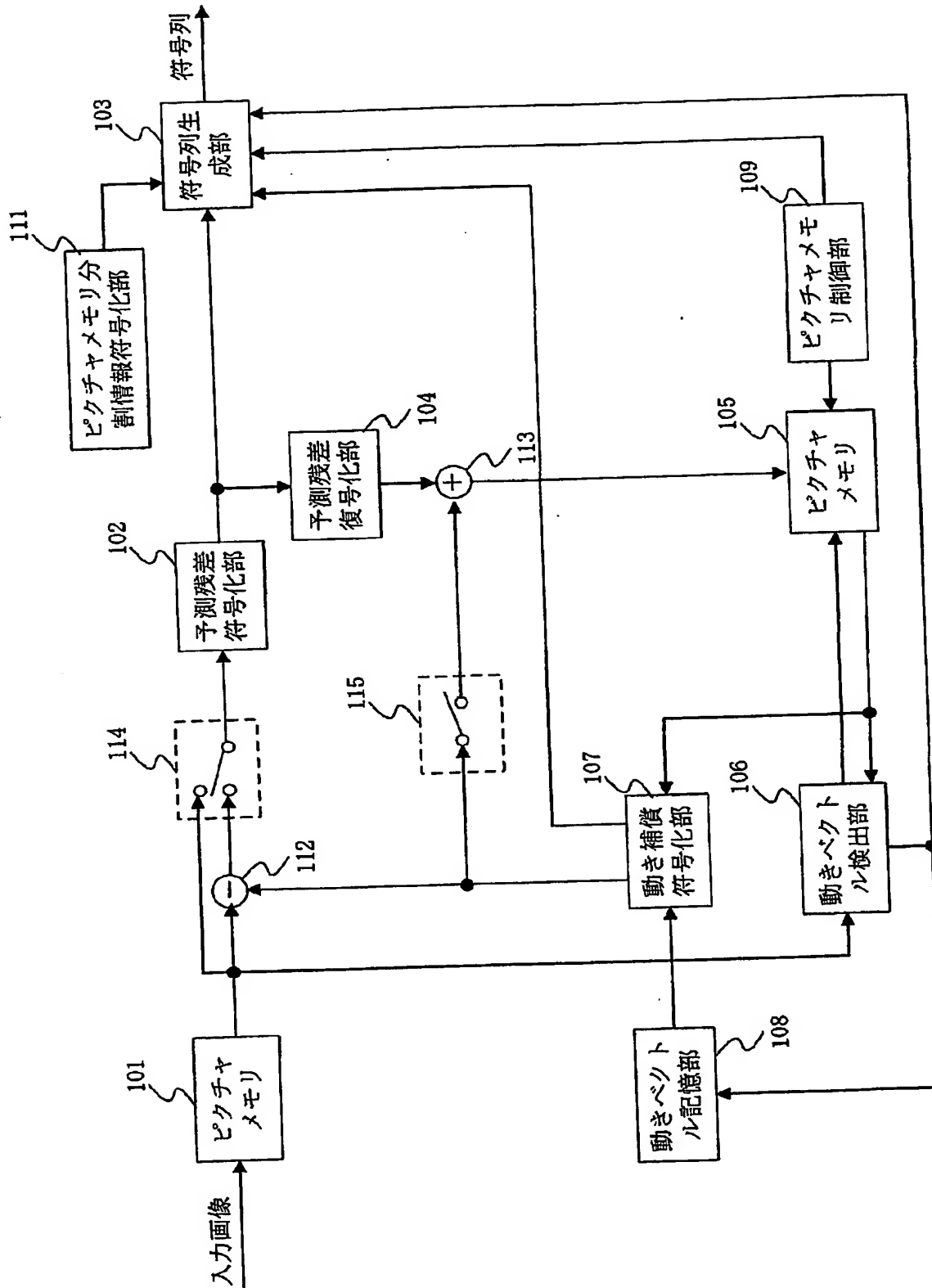
(b-1)



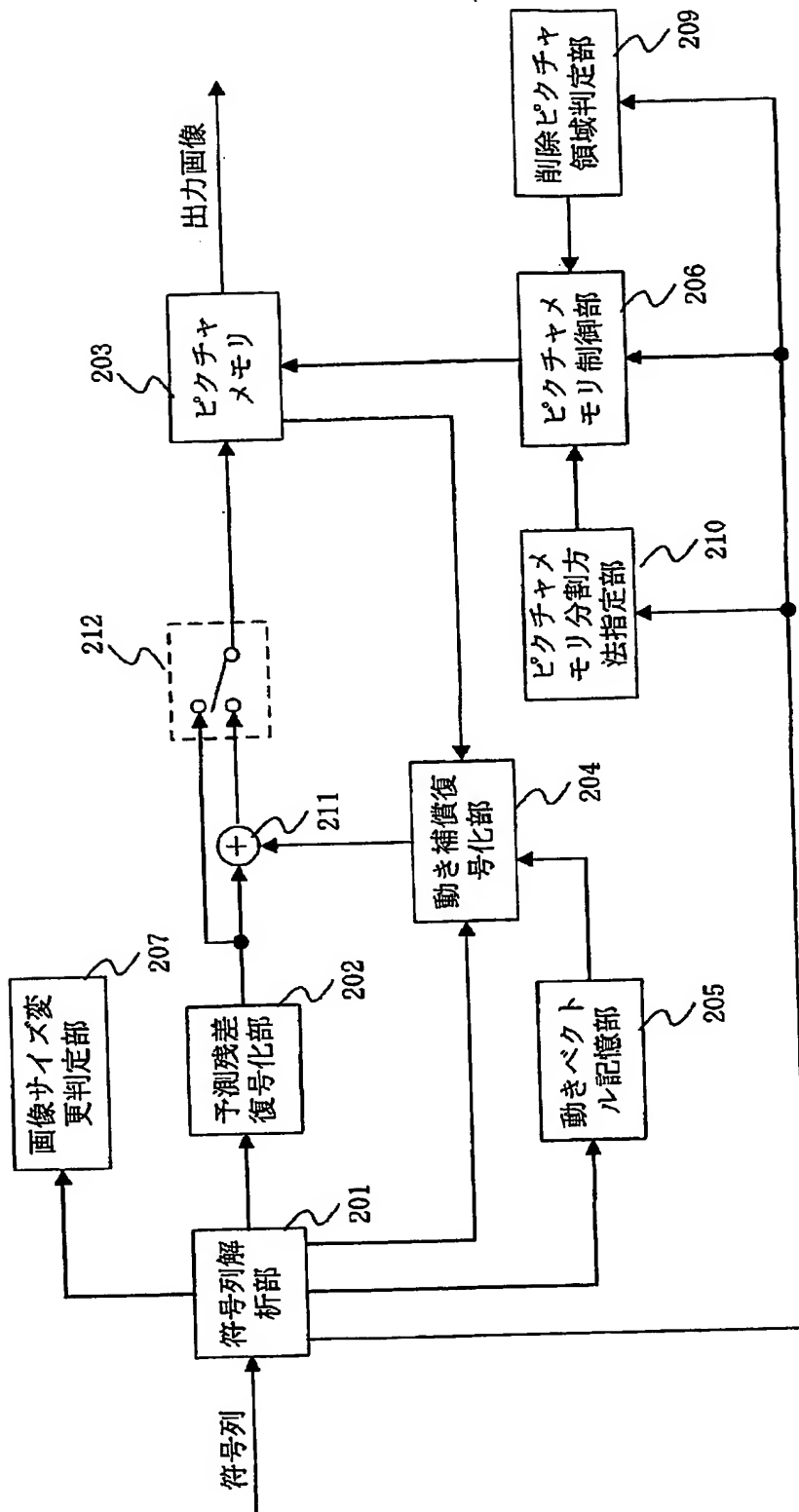
(b-2)



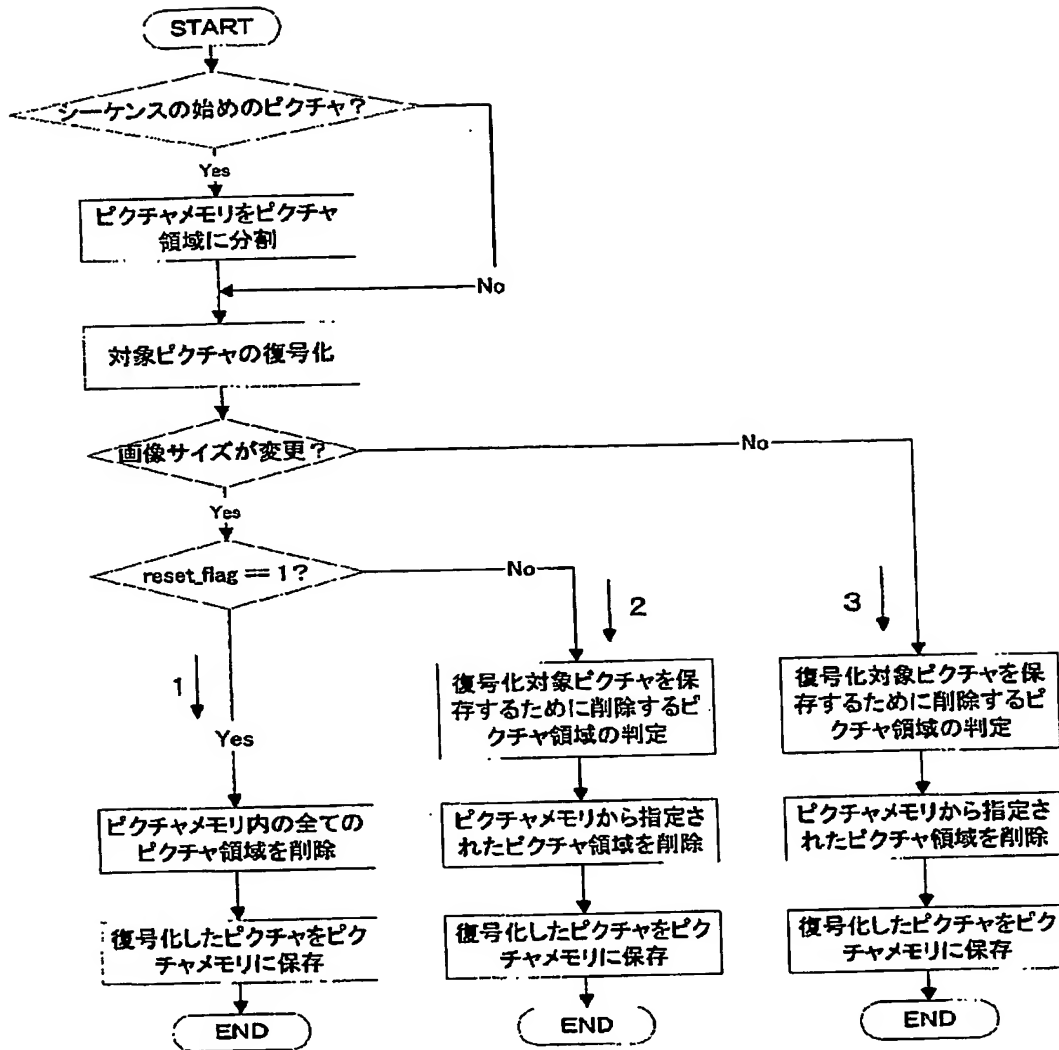
【図7】



【図8】

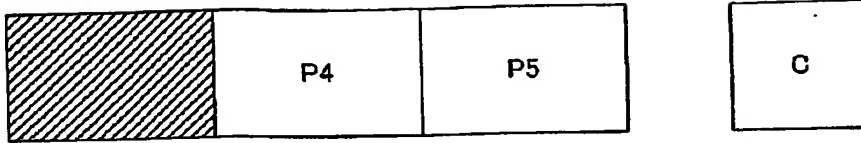


【図 9】

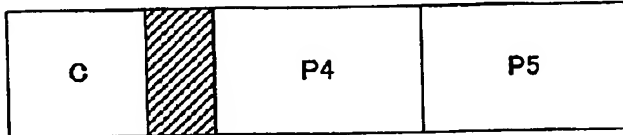


【図 10】

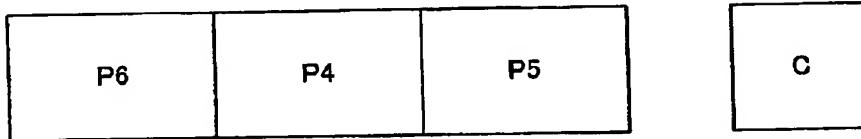
(a-1)



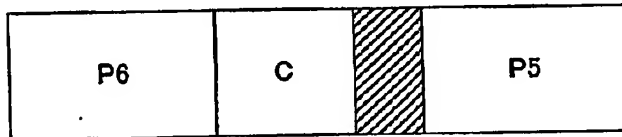
(a-2)



(b-1)

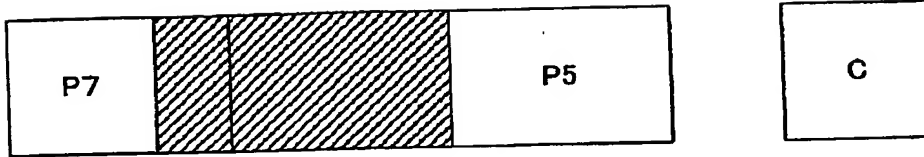


(b-2)

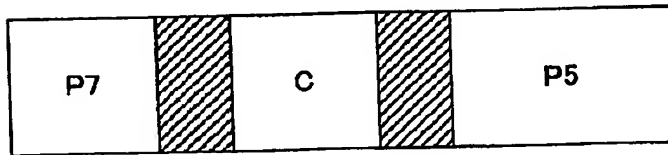


【図 11】

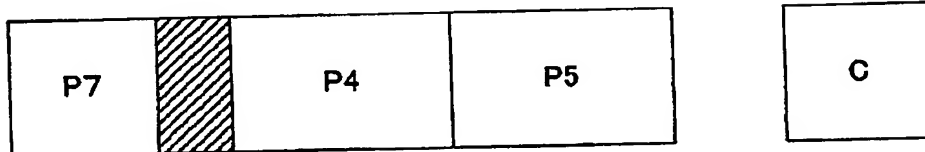
(a-1)



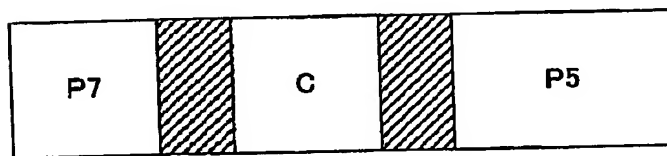
(a-2)



(b-1)

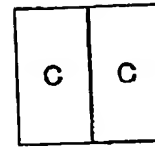
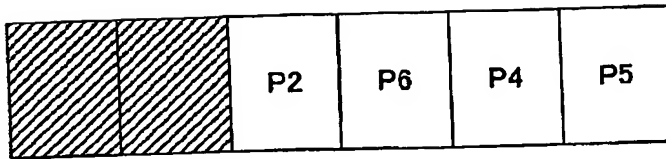


(b-2)

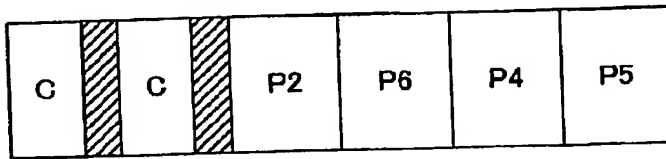


【図 12】

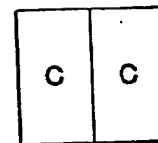
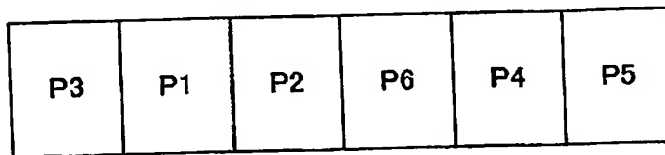
(a-1)



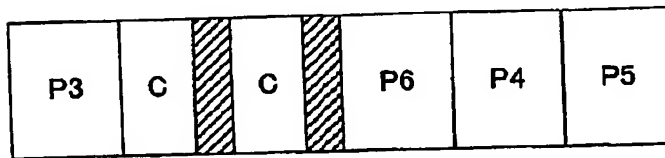
(a-2)



(b-1)

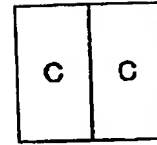
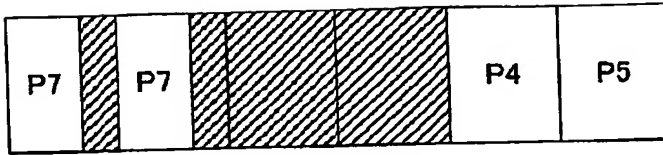


(b-2)

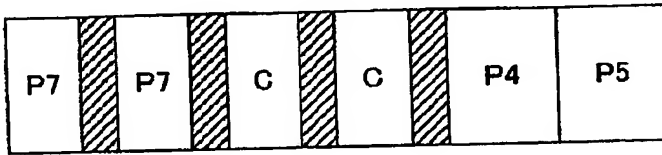


【図 13】

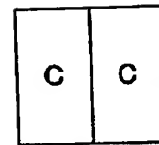
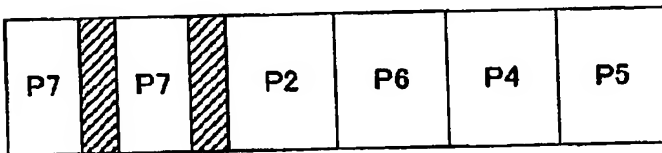
(a-1)



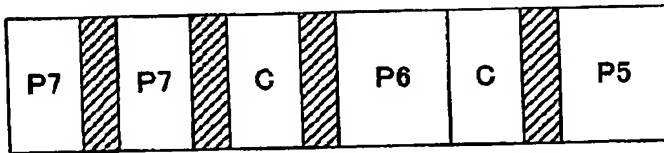
(a-2)



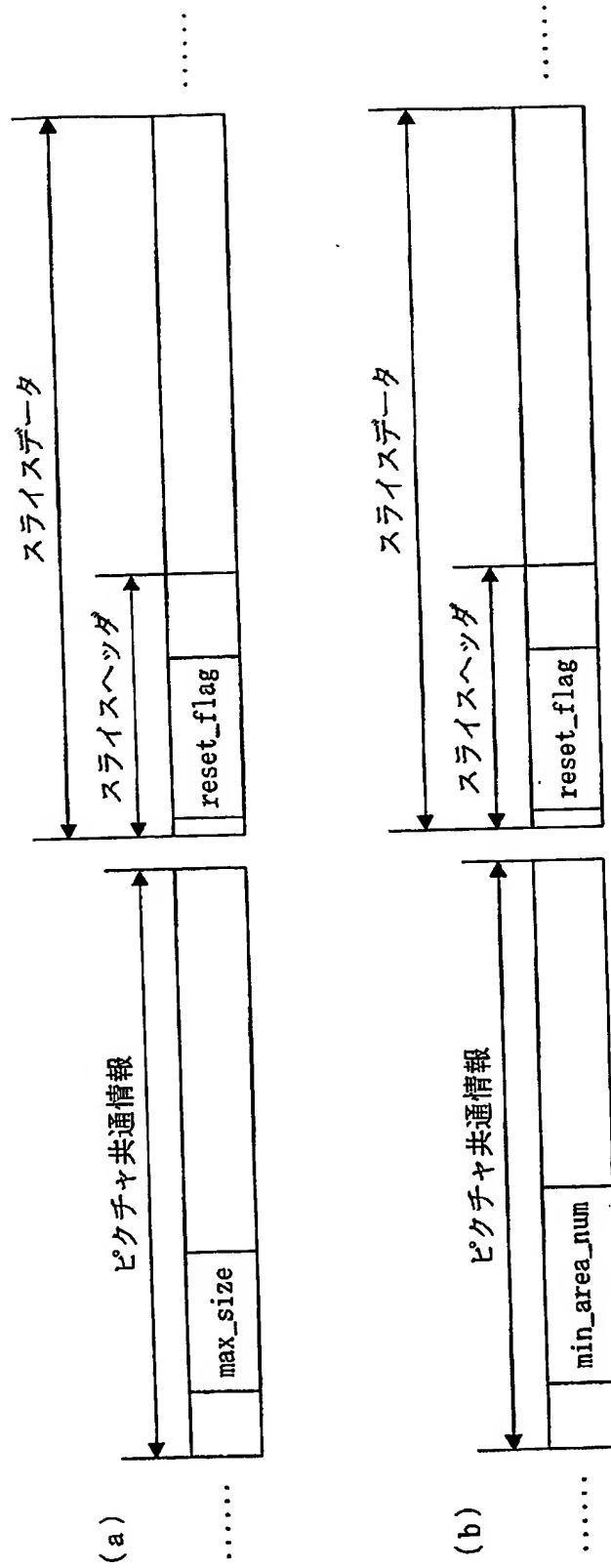
(b-1)



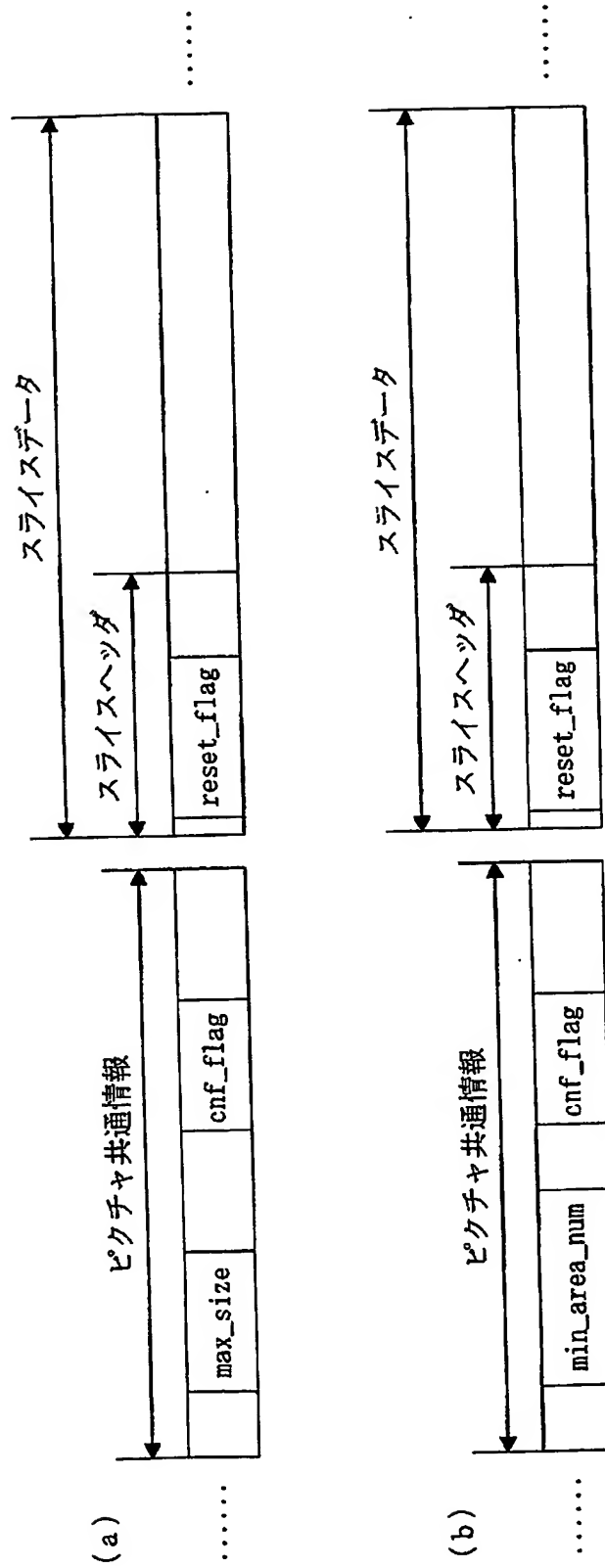
(b-2)



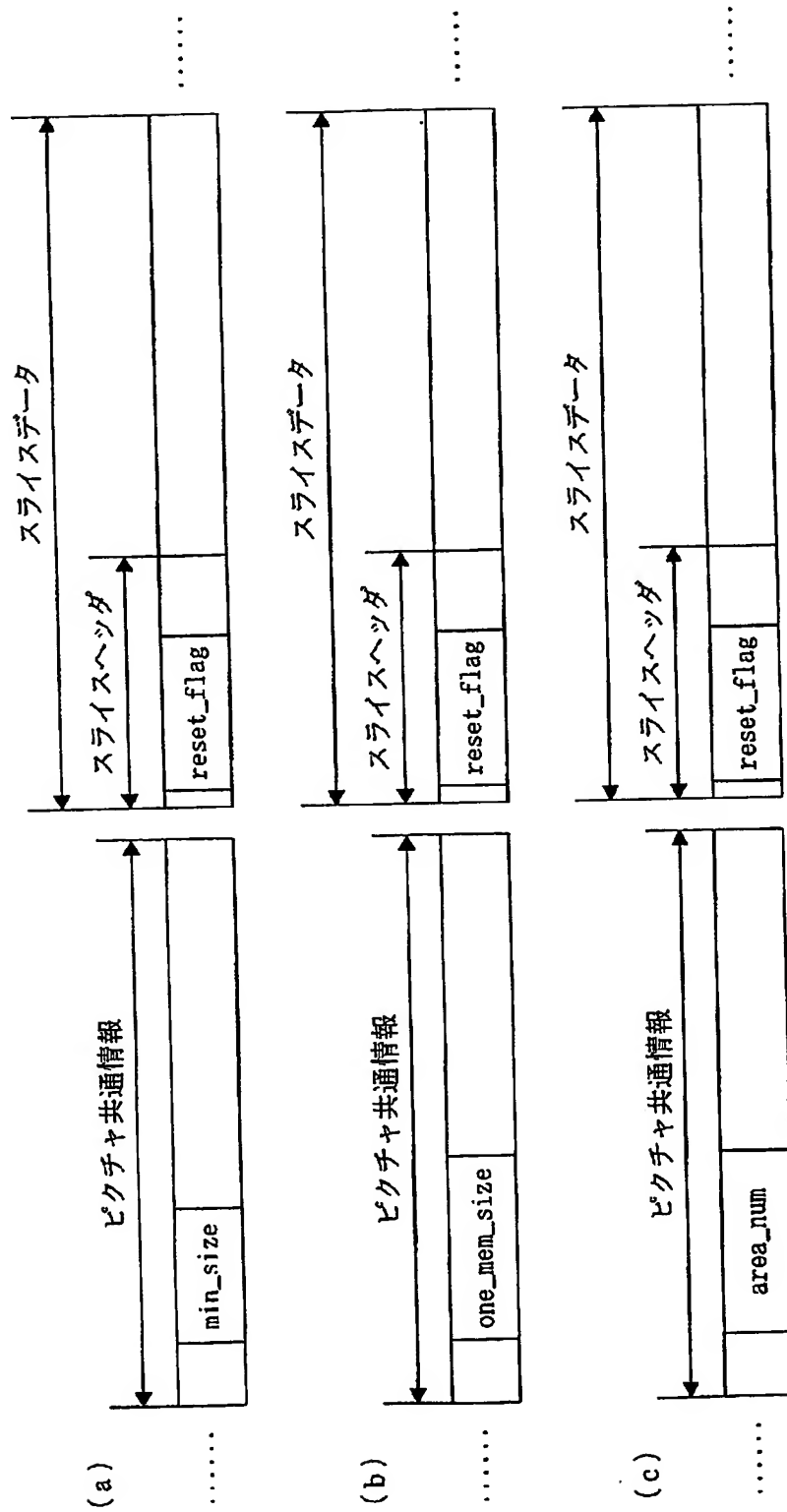
【図 14】



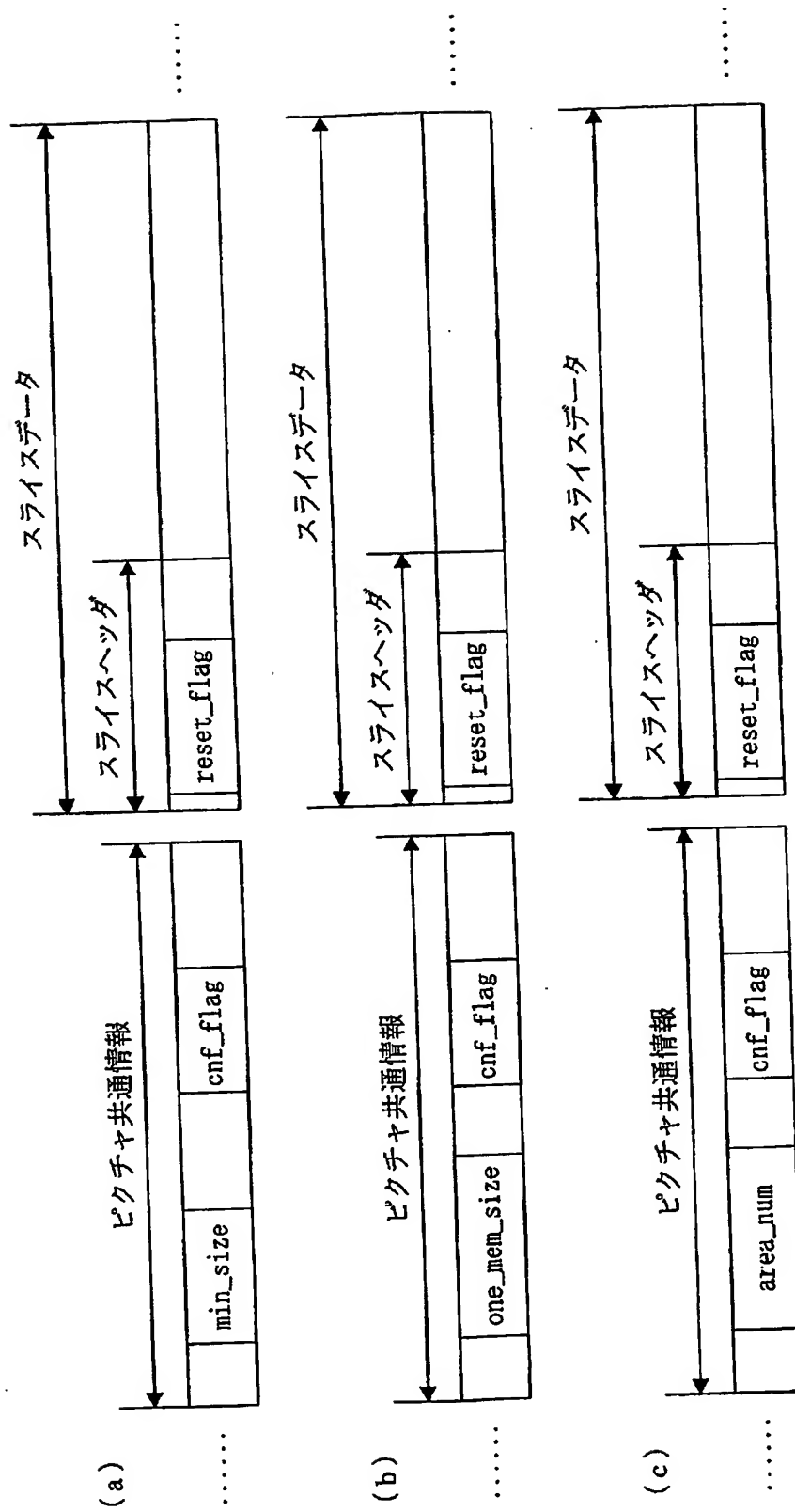
【図15】



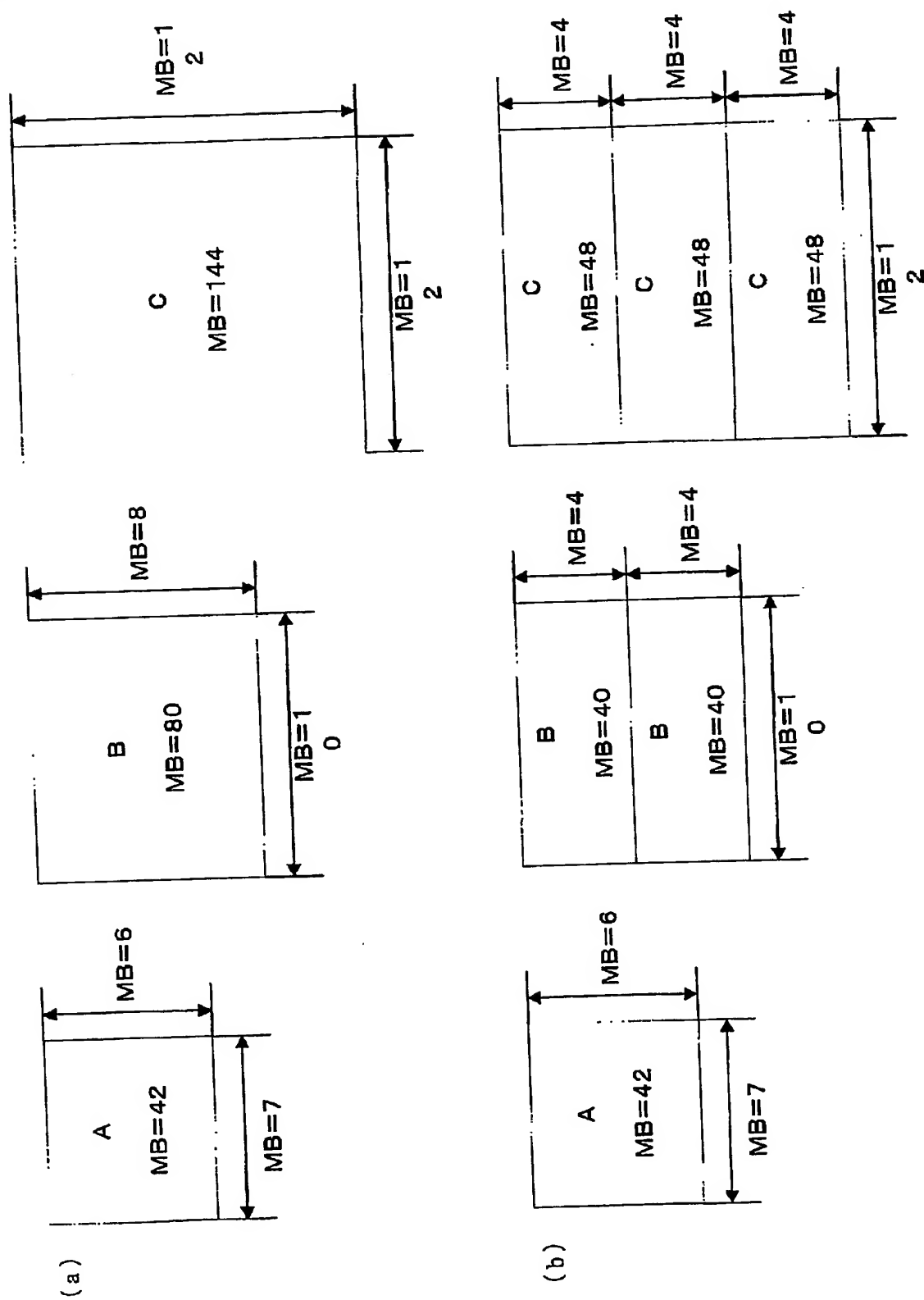
【図 16】



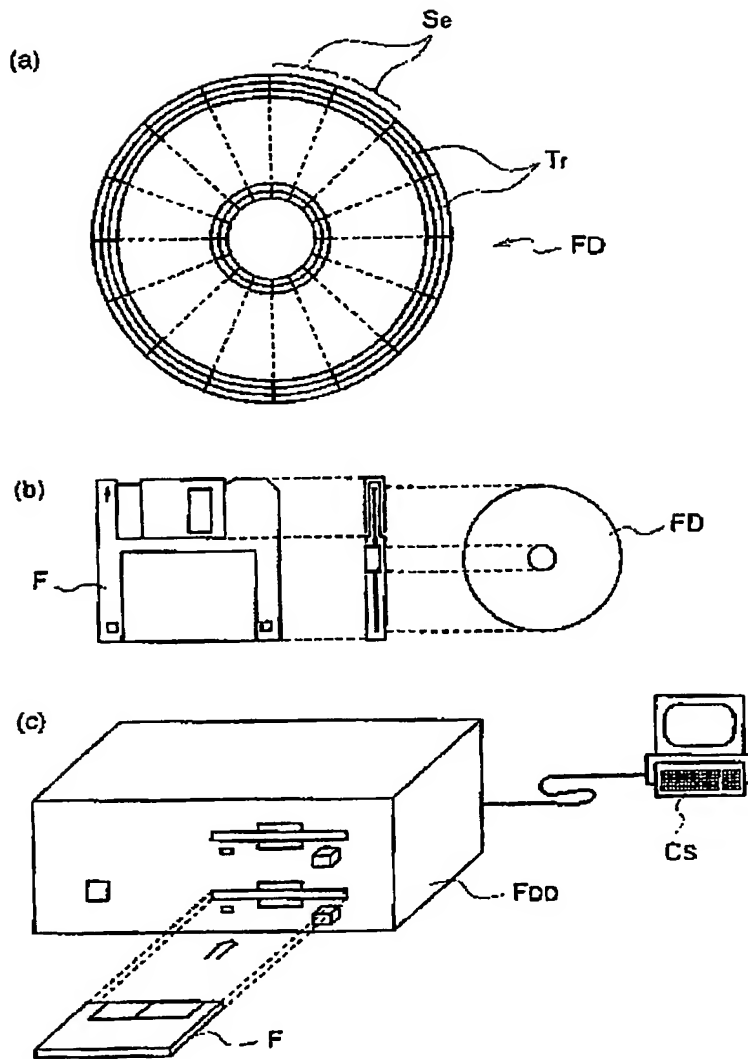
【図17】



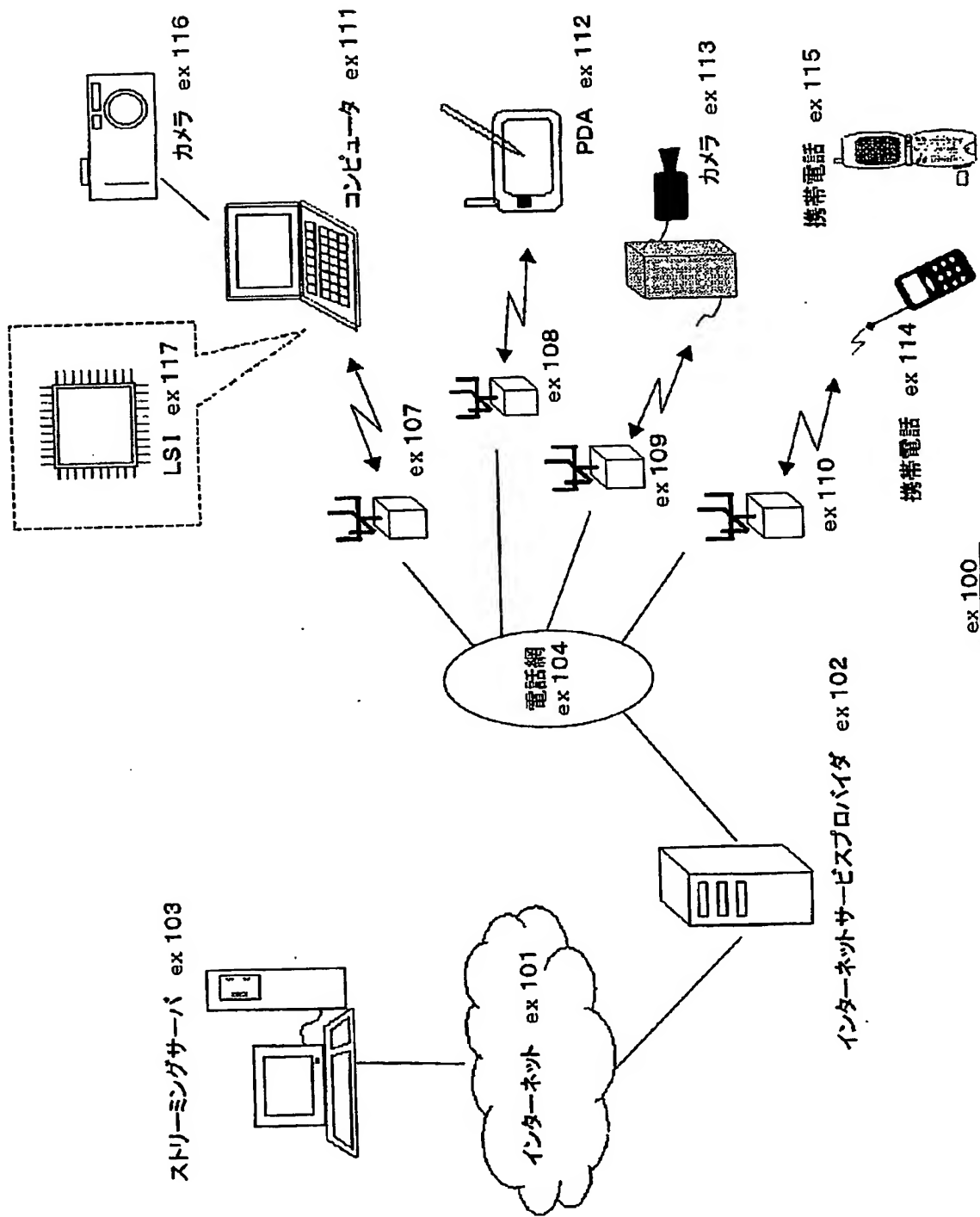
【図18】



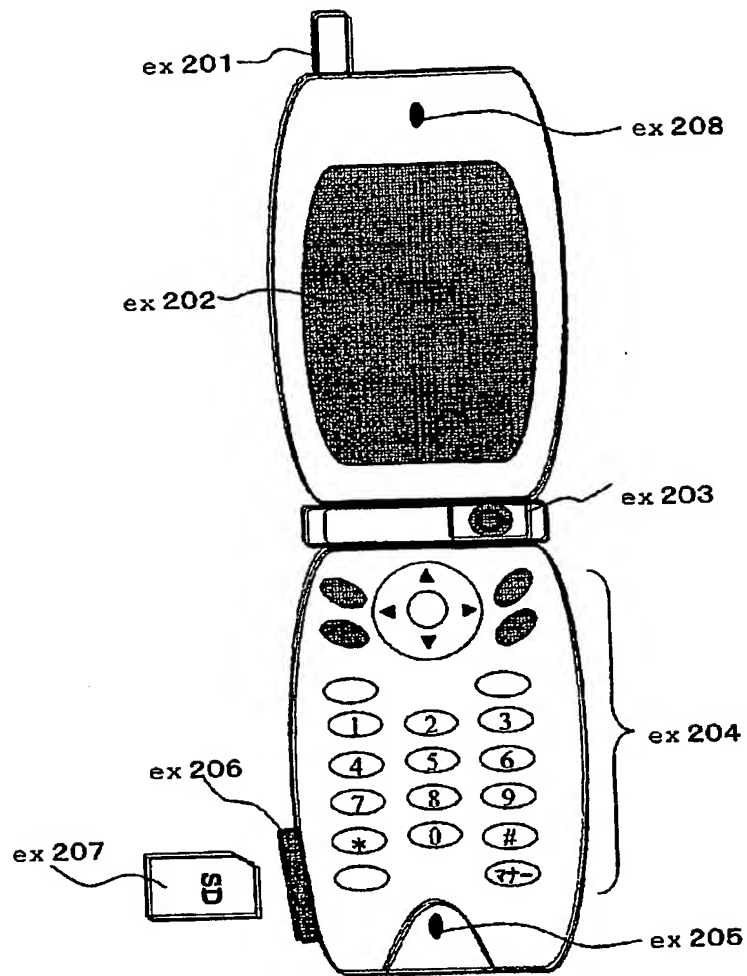
【図 19】



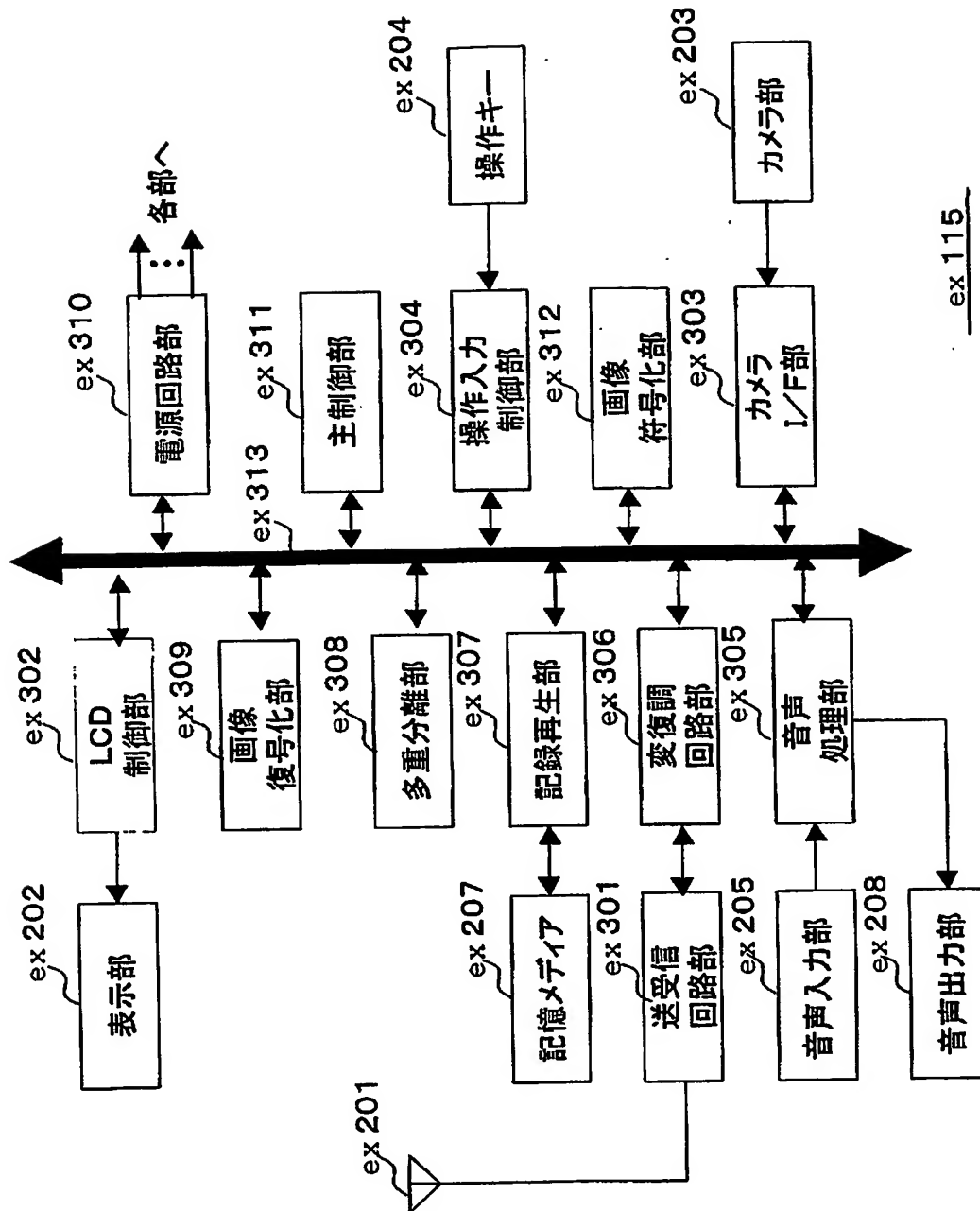
【図 20】



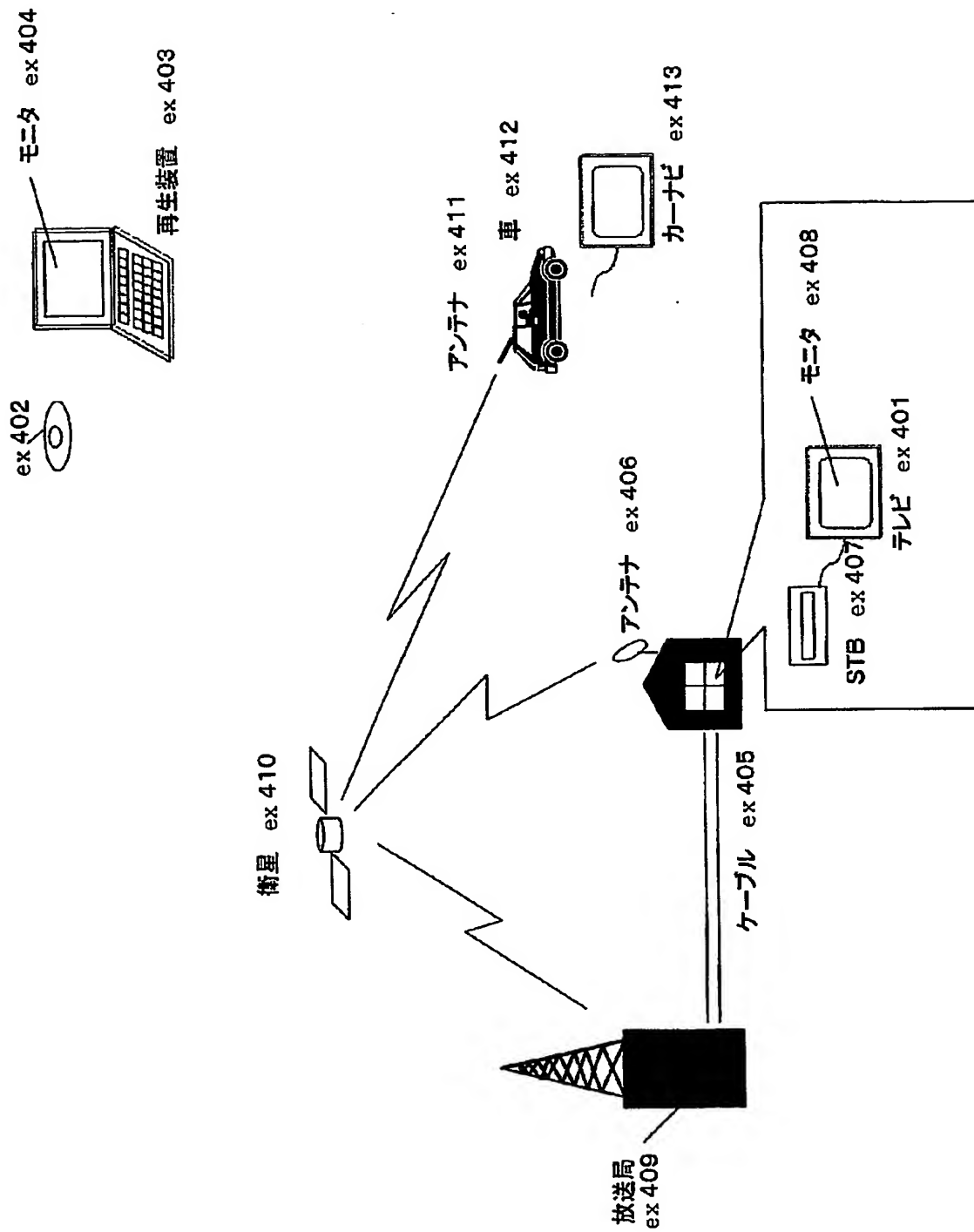
【図 21】



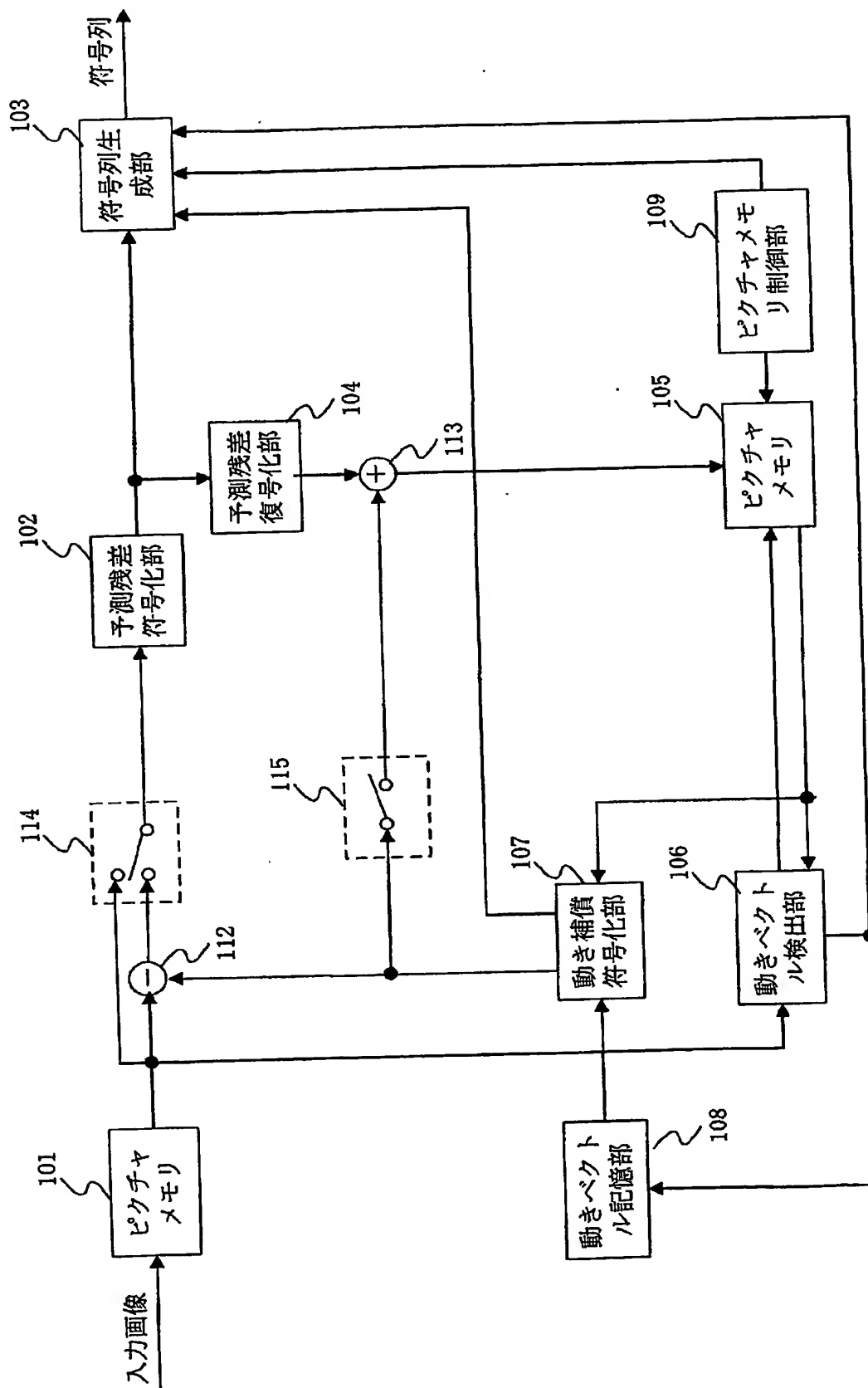
【図 22】



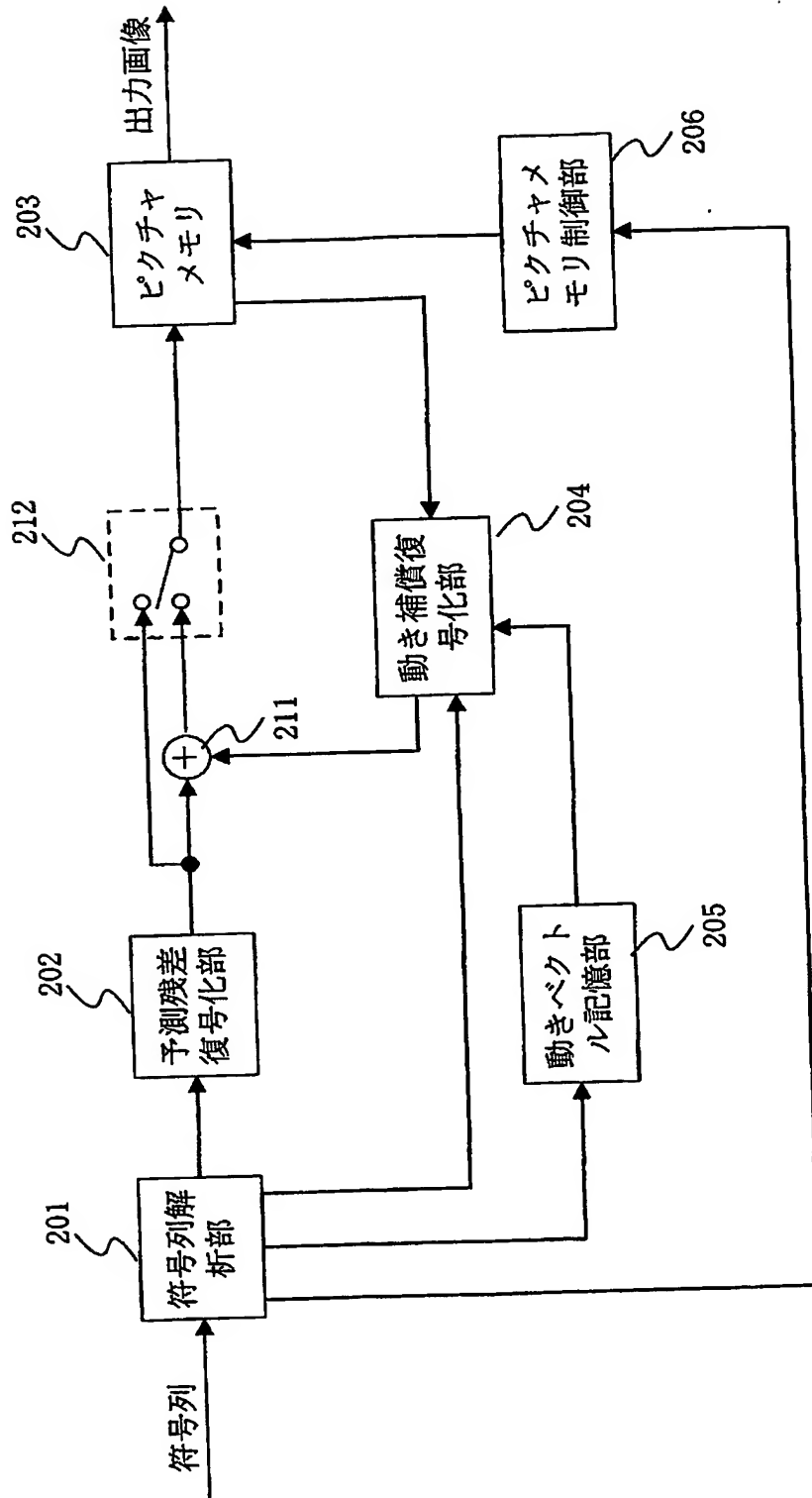
【図 23】



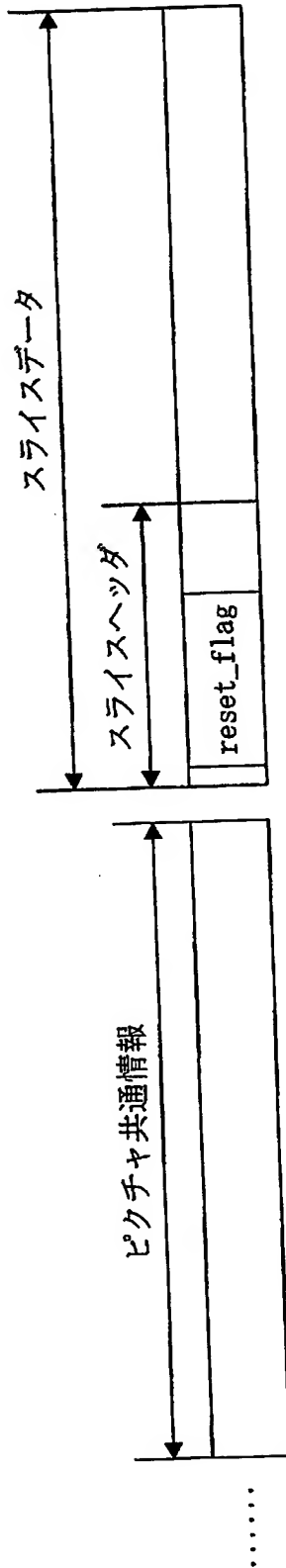
【図 24】



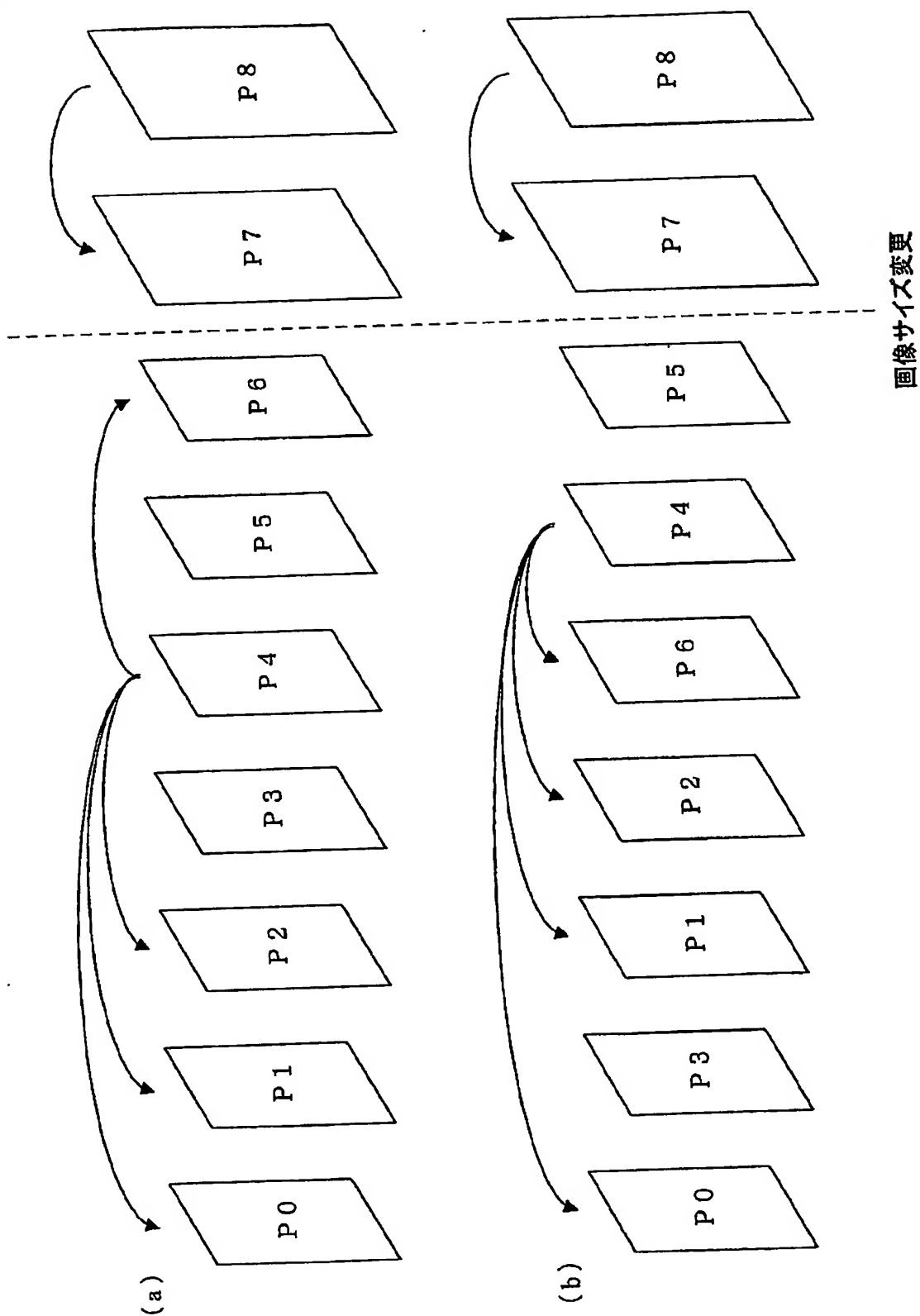
【図 25】



【図 26】



【図 27】



【書類名】 要約書

【要約】

【課題】 符号列の途中でピクチャの画像サイズが変更されたとき、復号化におけるピクチャメモリ内に複数の画像サイズのピクチャが混在しても、ピクチャメモリの管理を矛盾無く行うことを可能とする方法を提案する。

【解決手段】 符号化側において復号化側のピクチャメモリの管理処理を想定しながら、ピクチャメモリを分割するための情報を符号化する。復号化側では符号列中の前記情報を解析することにより、ピクチャメモリを1つもしくは複数の領域に分割し、前記分割された領域を1つの単位としてピクチャデータの削除および復号化されたピクチャの蓄積を行う。

【選択図】 図 8

認定・付加情報

特許出願の番号	特願 2003-040804
受付番号	50300262067
書類名	特許願
担当官	第一担当上席 0090
作成日	平成15年 2月20日

<認定情報・付加情報>

【提出日】	平成15年 2月19日
-------	-------------

次頁無

出証特 2004-3021932

特願 2 0 0 3 - 0 4 0 8 0 4

出 願 人 履 歴 情 報

識別番号

[0 0 0 0 0 5 8 2 1]

1. 変更年月日

1 9 9 0 年 8 月 2 8 日

[変更理由]

新規登録

住 所

大阪府門真市大字門真 1 0 0 6 番地

氏 名

松下電器産業株式会社